



**MESTERSÉGES INTELLIGENCIA ELEMekkel  
TÁMOGATOTT PROGRAMOZÁS OKTATÁSA**

**DOKTORI ÉRTEKEZÉS**

**PÁNTYA RÓBERT**

**2011**

Eötvös Loránd Tudományegyetem Informatika Doktori Iskola  
Az informatika alapjai és módszertana Doktori program

Iskolavezető: Dr. Benczúr András  
matematikai tudományok doktora, egyetemi tanár

Programvezető: Dr. Demetrovics János  
matematikai tudományok doktora, egyetemi tanár

Témavezető:  
Dr. Zsakó László  
tanszékvezető egyetemi docens

Média- és Oktatásinformatika Tanszék

Budapest, 2011

*Jutkának, Rékának és Nórának*

## Köszönetnyilvánítás

Köszönettel tartozom témavezetőmnek, **Dr. Zsakó Lászlónak** minden segítségéért, amellyel hozzájárult disszertációm elkészítéséhez. Értékes tanácsai, meglátásai, programozás-módszertani észrevételei nagyon hasznosak voltak számomra. Az a szemlélet, amelyet tőle tanultam, meghatározó számomra további pályám során is.

Köszönettel tartozom **Mucsics F. Lászlónak**, a Károly Róbert Főiskola adjunktusának, akivel együtt kezdtük az e-learning meghonosítását a Károly Róbert Főiskolán. Pedagógiai érke, tanítási, oktatási módszerei kutatásaimra nagy hatással voltak. Köszönöm segítségét a statisztikai elemzések kivitelezésében, a szakszerű fordítások elvégzésében és köszönöm minden kritikai észrevételét.

Köszönettel tartozom **Sinka Istvánnénak** a gyöngyösi József Attila Szakközépiskola, Szakiskola és Kollégium igazgatónőjének, **Dr. Papné Mátyus Zsuzsanna** igazgatóhelyettesnek, hogy lehetővé tették számomra a pedagógiai kísérlet lebonyolítását intézményükben. Külön köszönet illeti **Vinczéné Nagy Erzsébetet**, az informatika munkaközösség vezetőjét, akinek támogatása nélkül mind a kísérlet sikeres lefolytatása, mind pedig az osztállyal való munka nem lett volna ilyen zökkenőmentes.

Nagyon hálás vagyok **Váradi Istvánnak**, a gyöngyösi Vak Bottyán János Katolikus Műszaki és Közgazdasági Középiskola igazgatójának, valamint **Bátoriné Zaja Évának**, az informatika munkaközösség vezetőjének, hogy támogatták és segítették intézményükben a pedagógiai kísérlet lefolytatását.

Köszönöm a József Attila Szakközépiskola, Szakiskola és Kollégium **11.a, 12. a és 12.b osztályainak**, a Vak Bottyán János Katolikus Műszaki és Közgazdasági Középiskola **12.b és 13.c osztályainak**, valamint a Károly Róbert Főiskola **első évfolyamos Gazdaságinformatikus BSc. szakos hallgatóinak**, hogy a pedagógiai kísérletben részt vettek a kísérleti, vagy a kontroll csoportok tagjaiként. Külön köszönet illeti a kísérleti csoportok tagjait, akik három hónapon keresztül ismerkedtek a logikai programozás rejtelmeivel.



## Tartalomjegyzék

I.	Bevezetés .....	1
I.1.	A témaválasztás indoklása .....	2
I.2.	A kutatási téma előzményei .....	3
I.3.	Célkitűzések .....	4
II.	Mesterséges intelligencia főbb témaköreinek áttekintése .....	5
II.1.	A szakértői rendszerek áttekintése .....	7
II.2.	A Prolog nyelv jellemzőinek áttekintése .....	12
III.	Pedagógiai kihívások az Internet világában .....	17
III.1.	A lakosság info-kommunikációs eszközellátottsága .....	18
III.2.	Programozott oktatás használata a távoktatásban e-learning segítségével ...	21
III.2.1.	Távoktatás és programozott oktatás .....	21
III.2.2.	E-learning rendszerek jellemzői .....	24
III.3.	Megoldási javaslatok .....	27
IV.	Intelligens segédanyag az anyagszerkezet oktatásához .....	31
IV.1.	Az intelligens tananyag .....	32
IV.2.	Anyagszerkezeti alapok a feladatmegoldásokhoz .....	35
IV.3.	Feladatok kvantumszámokra .....	37
IV.4.	Feladatok elemek elektronszerkezetére .....	41
IV.5.	Feladatok az oxidációs számok és az elektronegativitás meghatározására ..	46
V.	Mesterséges intelligencia és Logikai programozás oktatása e-learning rendszerben .....	55
V.1.	A Mesterséges intelligencia oktatási lehetőségei e-learning rendszerben ....	56
V.2.	A Logikai programozás oktatási lehetőségei e-learning rendszerben .....	60
V.3.	Kapcsolat a Prolog nyelv és az SQL nyelv között .....	65
V.3.1.	Minta-adatbázis .....	66
V.3.2.	A minta-adatbázis Prolog nyelven .....	67
V.3.3.	Feladatok és megoldásuk SQL és Prolog nyelven .....	68
VI.	A kísérleti kurzusok eredményeinek, tapasztalatainak kiértékelése, elemzése ...	71
VI.1.	Bevezetés .....	71
VI.2.	A Logikai programozás kísérleti kurzus tananyagának bemutatása .....	73
VI.3.	A Logikai programozás kísérleti kurzus eredményeinek statisztikai elemzése .....	99

VI.3.1. Módszer.....	99
VI.3.2. Bemeneti tesztek értékelése .....	100
VI.3.2.1. József Attila Szakközépiskola, Szakiskola és Kollégium bemeneti tesztjei .....	100
VI.3.2.2. Vak Bottyán János Katolikus Műszaki és Közgazdasági Középiskola bemeneti tesztjei .....	101
VI.3.2.3. Károly Róbert Főiskola bemeneti tesztjei.....	103
VI.3.2.4. Mindhárom csoport bemeneti tesztjeinek összehasonlítása .....	104
VI.3.3. Kimeneti tesztek értékelése.....	105
VI.3.3.1. József Attila Szakközépiskola, Szakiskola és Kollégium kimeneti tesztjei .....	105
VI.3.3.2. Vak Bottyán János Katolikus Műszaki és Közgazdasági Középiskola kimeneti tesztjei .....	106
VI.3.3.3. Károly Róbert Főiskola kimeneti tesztjei .....	107
VI.3.3.4. Mindhárom csoport kimeneti tesztjeinek összehasonlítása .....	108
VI.3.4. A kísérlet értékelése .....	109
VI.3.5. A kísérlet alapján levonható következtetések .....	115
VII. Összefoglalás .....	117
VIII. Summary.....	118
IX. Irodalomjegyzék .....	119
X. A szerző disszertációhoz kapcsolódó publikációi.....	125
XI. Függelék.....	127
A függelék – Az intelligens tananyag forráskódja .....	127
B függelék - Programozott tananyag az atomok elektronszerkezetének meghatározására.....	144
C függelék – Gyakorló feladatsorok .....	163
D függelék – Bemeneti teszt .....	173
E függelék – A bemeneti tesztek eredményei .....	180
F függelék – Kimeneti teszt.....	186
G függelék – A kimeneti tesztek eredményei .....	194
H függelék – Középiskoláknak kiküldött kérdőív .....	200

## I. Bevezetés

A kutatás **oktatás-módszertani** jellegű, a problémamegoldó képesség fejlesztésére fókuszál. A problémamegoldásra és így a problémamegoldás fejlesztésére különösen alkalmasak a **logikai nyelvek** (pl. a Prolog), amelyek szabályok és tények alapján dolgoznak. Ezek a nyelvek a legfontosabb eszközei a mesterséges intelligenciát alkalmazó programozásnak és a szakértői rendszereknek is.

Kutatásom során a **problémamegoldó gondolkodás fejlesztésére** olyan hasznos esettanulmányokat, intelligens oktatási segédanyagokat készítettem, amelyek különféle természettudományos problémákat (pl.: az elemek elektronszerkezete, az elemek kötéstípusai, stb.) oldanak meg Prolog logikai programnyelv segítségével. Ezeknek a feladatoknak a „hagyományos” megoldása természetes intelligenciát igényel, melyet a kutatásom során egy logikai programozási nyelv segítségével oldottam meg.

A konkrét problémák számítógépes megoldása jó alapot nyújt a **logikai nyelvek** (pl. a Prolog) **oktatásának módszertani kérdéseihez** is, így a mesterséges intelligencia elemekkel támogatott programozás oktatását egy hatékony eszközzel tudom segíteni.

Az **intelligens oktatási segédanyagok** legnagyobb erénye abban rejlik, hogy arra ösztönzi a tanulókat, hogy ők is készítsenek ilyen szabályalapú rendszereket. Vagyis próbálják meg a tudásanyagot megfelelő formába önteni, majd értelmes kérdések után hasznos következtetésekre jutni. Ezeknek a segítségével pedig az egyik legfontosabb oktatási célunkat tudjuk elérni, vagyis a problémamegoldó képességüket tudjuk fejleszteni.

2008. tavaszától oktatok a Károly Róbert Főiskolán **Mesterséges intelligencia** tantárgyat. A kurzus során a hallgatók a mesterséges intelligencia tárgyköre mellett megismerhetik a szakértői rendszereket, betekintést nyerhetnek a robotikába, valamint a logikai programozással ismerkedhetnek meg.

A tantárgy oktatása során bevezetésre kerültek a kutatásom során elért eredmények. A hallgatókkal speciális szakértői rendszereket fejlesztünk a Prolog nyelv segítségével. Ezek az esettanulmányok nagyon hasznosnak bizonyulnak a problémamegoldó képességük fejlesztésére.

Az info-kommunikációs eszközök és az Internet széleskörű hazai elterjedése miatt különösen a távoktatásban, levelezős képzésben (de a nappalis képzésben is) **e-learning** segítségével célszerű támogatni az oktatást.

A mesterséges intelligencia (s ezen belül különösen a logikai programozás) oktatása **Moodle** e-learning rendszerrel kerül támogatásra. Ennek során nagy hangsúlyt fektetnek a **programozott oktatás módszertanát** követő leckék elkészítésére és hatékony felhasználására, valamint a **konstruktivista pedagógia** alkalmazására.

## **I.1. A témaválasztás indoklása**

A logikus gondolkodás tisztelete és szeretete számomra mindig nagyon fontos volt. A sakk, valamint a matematika fejlesztő hatását a problémamegoldó képesség fejlesztésében saját példámon is megtapasztaltam. Az egyetemi informatikai tanulmányaim alatt találkoztam a Prolog logikai programozási nyelvvel, amelynek különös gondolkodásmódja nagyon felkeltette az érdeklődésemet.

2001-től oktatok Gyöngyösön a Károly Róbert Főiskolán a Gazdaságmatematika és Informatika Tanszék oktatójaként. Az alapozó informatikai tárgyak mellett adatbáziskezelést és különböző programozási nyelveket tanítok. 2008. tavaszától Mesterséges intelligencia fakultatív tantárgyat is oktatok. A kurzus során a hallgatók a mesterséges intelligencia tárgyköre mellett megismerhetik a szakértői rendszereket, betekintést nyerhetnek a robotikába, valamint a logikai programozással ismerkedhetnek meg.

A Károly Róbert Főiskolán bekapcsolódtam a távoktatásos képzésbe. A távoktatás módszertani kérdéseit így közvetlenül is tudtam vizsgálni. Az e-learning szerteágazó módszertani fejlesztési lehetőségei felkeltették érdeklődésemet. Kollégámmal, Mucsics F. Lászlóval 2010-ben megalapítottuk a Károly Róbert Főiskolán az E-Learning Módszertani Egységet (ELME), melynek fő feladatai közé tartozik a főiskola e-learning-es kurzusainak megvalósítása, támogatása, e-learning-es projektek kivitelezése, valamint szakmai tanácsadás a kollégáink részére.

## **I.2. A kutatási téma előzményei**

1. Rendszeresen részt vettem különböző informatikai oktatás-módszertani, e-learning, illetve mesterséges intelligencia szakmai konferenciákon (Informatika Szakmódszertani Konferencia – INFODIDACT, Informatika a felsőoktatásban – IF, Matematikát, fizikát és informatikát oktatók konferenciája – MAFIOK, MOODLEMOOT konferencia, Intelligens Rendszerek – Fiatal Kutatók Szimpóziuma - IRFIX).
2. Gyöngyösön, Heves megyében 2004 óta minden évben készítem a Megyei Számítástechnika Alkalmazói Verseny, valamint a Városi Számítástechnika Alkalmazói Verseny versenyfeladatait, illetve ellátom a zsüri elnöki teendőit. Ennek során közvetlen tapasztalatokat szereztem a középiskolás diákok informatikai tudásáról, problémamegoldó képességéről. Az iskolákkal való aktív közreműködés eredményeként számos középiskola informatikai munkaközösségével sikerült jó kapcsolatokat kialakítanom.
3. A Károly Róbert Főiskolán bekapcsolódtam a hallgatók távoktatásos képzésébe. Mindig foglalkoztatott, hogy milyen módon lehetne fejleszteni ezt a képzést. Módszertani vizsgálódásokat folytattam a hatékonyabbá tétel érdekében.

4. Oktatói munkám során évről évre azt tapasztaltam, hogy az újabb és újabb hallgatói generációk, mind a középiskolákban, mind pedig a felsőoktatásban egyre kevésbé preferálják a természettudományos tárgyakat. Különösen érdekelt, hogy hogyan, milyen formában lehetne motiválni, érdeklődésüket felkelteni ezen tárgyak iránt.

### **I.3. Célkitűzések**

1. A természettudományos tárgyak népszerűsítése olyan intelligens oktatási segédanyagok elkészítésével, amelyek segítségével a természeti törvények könnyebben tanulhatókká válhatnak.
2. Olyan e-learning-ben támogatott kurzusokat kialakítani, amelyek segítségével a Mesterséges intelligencia, a Logikai programozás hatékonyabban taníthatóvá/tanulhatóvá válik. Olyan tanulók számára elérhetővé, könnyen befogadhatóvá tenni a logikai programozást, akik sohasem gondolták, hogy képesek lesznek ilyet tanulni.
3. A távoktatás számára olyan módszertani ajánlásokat tenni az új technológiai vívmányok (e-learning, web 2.0) segítségével, amelyekkel hatékonyabbá lehet tenni az eddigi oktatási/tanulási folyamatokat.
4. A tanulók problémamegoldó képességének fejlesztése logikai programozás oktatásával, blended learning elrendezésben. Pedagógiai kísérlet végzése a hatás mértékének a mérésére.

## II. Mesterséges intelligencia főbb témaköreinek áttekintése

A Mesterséges Intelligencia kifejezést John McCarthy hozta létre, aki 1956-ban a dartmouthi konferencián (Darthmouth College) használta ezt a fogalmat. Ezen a konferencián, melynek témája az emberi gondolkodás folyamatának gépi modellezése volt, foglalkoztak először ezzel a területtel. [1]

A mesterséges intelligencia fiatal tudomány, mintegy 60 éves múltra tekint vissza. Ezen rövid időszak azonban a kutatók szemlélete alapján több korszakra is osztható.

A mesterséges intelligencia tudományának deklarációjakor nagyratörő terveket fogalmaztak meg a kutatók. A kutatás fő irányának az általános célú problémamegoldó módszerek kidolgozását tekintették. Azonban a túl általános problémák megfogalmazása miatt nem érték el átütő eredményt.

A második korszakban már számoltak azzal, hogy nem lehet általános célú hatékony programokat készíteni. Ekkor már olyan szűkített feladatosztály megoldását tűzték ki célul, amelyben fel kellett használni a feladatra jellemző speciális ismereteket. Ennek a szemléletváltásnak köszönhetően ennek a korszaknak a programjai már jóval hatékonyabbnak bizonyultak.

A mai napig is tart a mesterséges intelligencia kutatás azon korszaka, amikor a kutatók erősen speciális célú programokat készítenek nagy mennyiségű szakértői ismeretanyag felhasználásával. Ezek az elkészült programok már a gyakorlatban is hasznosítható szakértői rendszerek, melyek sikerét megalapozza a rendszerek összefüggéseinek pontosabb feltárása, elemzése valamint használhatóságuk az oktatásban. [2]

A Mesterséges intelligencia céljaként S. J. Russell és P. Norvig [3] alapján az alábbi, négyféle rendszer létrehozása fogalmazható meg:

- Emberi módra gondolkodó rendszerek
- Racionálisan gondolkodó rendszerek
- Emberi módra cselekvő rendszerek
- Racionálisan cselekvő rendszerek

A mesterséges intelligencia (MI) fogalmának általánosan elfogadott definíciója még mindig hiányzik. Az előbbi besorolás szerint a tankönyvekben az alábbi meghatározások, definíciók léteznek erre a fogalomra.

#### **Emberi módra gondolkodó rendszerek**

- *„Az MI egy izgalmas erőfeszítés a számítógépek gondolkodóvá tételére, értelemmel bíró gépek létrehozására a szó szoros értelmében.” [4]*
- *„Az MI az emberi gondolkodáshoz asszociált tevékenységek, mint a döntéshozatal, problémamegoldás, tanulás automatizálása vizsgálata.” [5]*

#### **Racionálisan gondolkodó rendszerek**

- *„Az MI a mentális képességek tanulmányozása számítógépes modellek segítségével.” [6]*
- *„Az MI az érzékelést, gondolkodást és cselekvést lehetővé tevő számítások tanulmányozása.” [7]*

#### **Emberi módra cselekvő rendszerek**

- *„Az MI olyan funkciók megvalósítására alkalmas gépek megalkotásának tudománya, mely funkciókhoz intelligenciára van szükség, amennyiben azokat emberek valósítják meg.” [8]*
- *„Az MI annak tanulmányozása, hogyan lehet számítógéppel olyan dolgokat tenni, melyeket jelenleg az emberek jobban tudnak.” [9]*

#### **Racionálisan cselekvő rendszerek**

- *„Az MI olyan kutatási terület, amely számítási eljárásokkal próbálja magyarázni és utánozni az intelligens viselkedést.” [10]*
- *„Az MI a számítástudomány azon ága, mely az intelligens viselkedés automatizálásával foglalkozik.” [11]*



A mesterséges intelligencia főbb alkalmazási területei vázlatosan:

- Szakértő rendszerek,
- Mesterséges látás,
- Természetes nyelvmegértés,
- Beszédmegértés,
- Gépi tanulás,
- Tervezés, robotika,
- Neurális hálózatok,
- Programozási nyelvek, stb.

A továbbiakban a mesterséges intelligencia azon két területét fejtem ki még részletesebben (szakértői rendszerek, programozási nyelvek), amelyek a dolgozatom alapját is képezték.

## II.1. A szakértői rendszerek áttekintése

Borgulya [12] így határozza meg a szakértői rendszer fogalmát:

*„A szakértői rendszer a tudásalapú rendszerek közé tartozik és jól strukturálható, könnyen formalizálható, szabályokkal leírható ismeretanyag esetén alkalmazható. Fő feladata a problémamegoldás, és ezt az emberi problémamegoldáshoz hasonló módon próbálja megvalósítani.”*

A szakértői rendszereket olyan szűk problémakör megoldására alkották, alkotják, amelyek megoldásához intelligenciára van szükség, így emberi szakértők szükségessége hozzá.

Mallach [13] a következőképpen foglalja össze a gépi szakértőnek az emberhez viszonyított **előnyös** és **hátrányos** tulajdonságait.

#### **A szakértői rendszerek főbb előnyei:**

- Több szakértő tudását integrálják magukba, ezáltal jobb döntéseket hozhatnak, mint az emberi szakértő.
- Gyorsabban adnak megoldást az embernél.
- A problémamegoldás eredménye mindig ugyanaz, függetlenül a külső körülményektől.
- Egymásután többféle problémát is meg tudnak oldani. Az átváltás egyik problémáról a másikra pillanatok műve.
- Alkalmazásuknak nincsenek helyi, időbeli korlátai.

#### **A szakértői rendszerek főbb hátrányai:**

- Csak egy szűk szakterületen használhatóak, amelyre fejlesztették őket.
- Csak a megadott szabályok szerint tudnak gondolkodni.
- Nem veszik észre határaikat, szemben az emberrel, aki észleli, ha tudása határához ér.

A szakértői rendszerek által kezelt problémátípusok P. Harmon és B. Sawyer [14] szerint a következőképpen csoportosíthatóak:

- procedurális,
- diagnosztizáló,
- monitorozó vagy őrző,
- konfiguráló vagy objektumtervező és
- tevékenységtervező vagy ütemező problémák.

Sántáné Tóth E. [15] szerint az egyes problémátípusok az alábbiak szerint jellemezhetőek.

A **procedurális** problémák esetén az adott tárgyköri ismeretanyag leírható valamely egyszerű eljárással, ilyenkor a probléma egymás utáni lépések sorozatával (algoritmikusan) leírható, vagy döntési fába rendezhető.

A **diagnosztizáló** problémák megoldása a megfigyelések halmaza fölötti következtetés egy vagy több lehetséges javaslatra. A **monitorozó** vagy **őrző** problémák során a jelek folyamatos figyelése történik, valamint akcióba lépés, ha a kívánttól eltérő szituáció áll elő, vagy hibás jelminta adódik.

A **konfiguráló** vagy **objektumtervező** problémák esetén a megfigyelések és a felhasználói igények alapján maga a rendszer dolgoz ki egy vagy több lehetséges megoldást, javaslatot az objektumok kapcsolatára vonatkozó megszorítások ismeretében. A **tevékenységtervező** vagy ütemező problémák az eléjük kitűzött cél érdekében tevékenységek olyan sorozatát határozzák meg, melyeknek az adott körülmények között történő végrehajtása célhoz vezet.

Sántáné Tóth E. [15] így fogalmaz a tudásról:

*„A tudás, az ismeret kulcsfontosságú a mesterséges intelligencián belül. A tudás sokrétű, nehéz pontosan jellemezni és formalizálni, szimbolizálni, továbbá állandóan változik.”*

B. C. Smith [16] alapján a tudásrepresentáció hipotézise azt mondja ki, hogy egy feladat megoldására csak olyan ismereteket tudunk mozgósítani, amelyet korábban reprezentáltunk, és amelyhez van egy azt olvasni, illetve feladatmegoldásra felhasználni tudó következtető módszerünk.

A leggyakrabban alkalmazott tudásrepresentációs módszer: a **szabályalapú tudásrepresentáció**. Tényállítások és HA-AKKOR (IF – THEN) szerkezetű feltételes állítások fogalmazhatók meg segítségükkel. Így a köznapi gondolkodás modellezésére, a szakértő tapasztalatait megfogalmazó **heurisztikák** leírására nagyon alkalmas.

Kétféle következtetési módszerrel dolgozhat egy szabályalapú rendszer: **célvezérelt** következtetéssel, valamint **adatvezérelt** következtetéssel.

Sántáné Tóth E. [15] így jellemzi a célvezérelt, illetve az adatvezérelt rendszereket:

*„Egy célvezérelt, visszafelé haladó rendszer egy feltételezett célállapotból visszafelé elindulva **AKKOR-HA szabályok** alkalmazásával megpróbálja a cél igazolását visszavezetni a rendszer által ismert tényekre, vagy korábban igazolt állításokra. Eközben, ha az adott cél (ill. közbülső rész cél) még igazolásra szorul, a rendszer keres egy ezzel illeszthető következményű szabályt, és az annak feltételében szereplő állítások (mint részcélok) igazolására vezeti vissza a munkát. Ha eközben zsákutcába jut, visszalépést alkalmaz, és új irányban próbálkozik.”*

„Egy adatvezérelt, előre felé haladó rendszer a kezdőállapotból elindulva valamely célállapot elérése érdekében **HA-AKKOR szabály** alkalmazásával halad előre. Mindaddig, míg célhoz nem ér, keres egy olyan szabályt, amelynek feltétele a munkamemória pillanatnyi tartalma felett igazolható, és végrehajtja az annak következményében leírt akciókat. Amennyiben egyszerre több szabály is alkalmazható lenne, konfliktus keletkezik, melyet valamely stratégia alkalmazásával felold.”

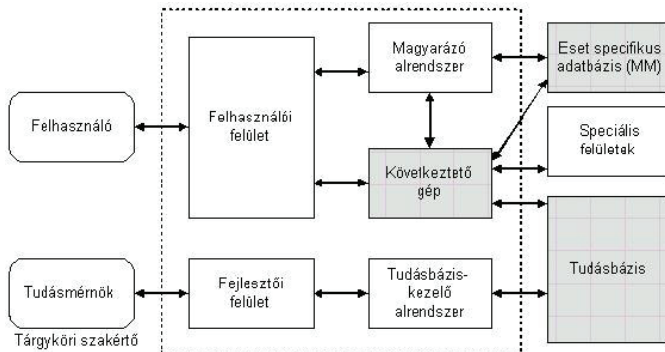
#### A szakértői rendszerek főbb részei:

- a tudásbázis, mely szabályok, objektumok, tények, heurisztikák segítségével tartalmazza a problémakör ismeretanyagát,
- a következtető gép, mely a tudásbázisból újabb tényeket vezet le,
- a magyarázó alrendszer, amely megmagyarázza a kérdéseket, megindokolja a problémára adott válaszokat,
- a dialógus komponens (felhasználói felület) és
- ismeretszerző komponens (fejlesztői felület), amelyek különböző célból kommunikálnak a felhasználóval, illetve a fejlesztővel. [15]

A szakértői rendszer működéséhez két alapvető elem (a tudásbázis és a következtető rendszer) minden esetben szükséges. Az 1. ábra szemlélteti a szakértői rendszerek felépítését, részeit.

1. ábra

#### A szakértői rendszerek felépítése, részei



Forrás: [15]

Azokat a fejlesztőeszközöket nevezik szakértői rendszer-kereteknek, vagy szakértői rendszer shelleknek, melyek felépítése hasonló az 1. ábrán ismertetettel, viszont tudásbázisuk üres. A shell tudásbázisának feltöltése után működőképes szakértői rendszer kapható. Az 1. táblázat különböző szakértői keret-rendszereket részletez.

**1. táblázat**

**Szakértői keret-rendszerek**

MProlog Shell	MProlog nyelvű tudásbázist kezelő eszköz.
ALL-EX Plus	Tudásalapú szimulációt támogató CS-Prolog alapú eszköz.
GENESYS	Szabályalapú eszköz főként osztályozó, diagnosztikai típusú rendszerek fejlesztéséhez.
CLIPS, Goldworks, G2, LEvel5	Egyéb szakértői keret-rendszerek.

Forrás: [15], [17]

A következő táblázatok néhány fontosabb, gyakorlatban megvalósult szakértői rendszert foglalnak össze az orvostudomány, kémia, informatika, térinformatika, üzleti élet és egyéb szakterületekről. [17]

**2. táblázat**

**Orvosi szakértői rendszerek**

CARDIAG-2/RHEUMA	Reumás betegségek diagnosztizálására.
MEDWISE	Orvosi diagnosztikák segítésére.
CASNET	Zöldhályog betegségben szenvedő betegek diagnosztizálására, kezelésére.
ESTROPID	Trópusi betegségek megállapítására.
NES	Fejlődésneurológiai szakértő rendszer.
MYCIN/NEOMYCIN	A vér bakteriális eredetű fertőzései, agyhártyagyulladás diagnosztizálására, terápia meghatározására.
INTERNIST	Orvosi rendszer belgyógyászati betegségek diagnosztikájának támogatására.
GUIDON/GUIDON2	Intelligens támogatás orvostanhallgatók oktatásában.
QMR (Quick Medical Reference)	Elektronikus kézikönyv.

Forrás: [1], [12], [17], [18]

3. táblázat

## Kémiai szakértői rendszerek

DENDRAL (Dentritic Algorithm)	Ismeretlen szerves vegyületek molekulaszervezetének meghatározására tömegspektrográfiai és mágneses rezonancia mérési adatokból.
MetabolExpert	Kémiai, orvosi, biológiai előrejelzésekhez.
PIRExS	Szervetlen kémiai reakciók előrejelzésére.

Forrás: [1], [12], [15], [17], [18]

4. táblázat

## Informatikai, térinformatikai szakértői rendszerek

R1 XCON	VAX számítógéprendszerek konfigurálásának támogatására.
CAPE	Számítógépes protokollok elemzéséhez.
OPSQL	Oracle alapú intelligens adatszótár.
HEARSEY	Angol beszédmegértő rendszer.
PROSPECTOR	Geológusok segítése ásványlelőhelyek felkutatásában.

Forrás: [1], [12], [15], [17]

5. táblázat

## Az üzleti élet szakértői rendszerei

CREDITEXPERT	Hitel elbírálásához.
PANGAEA	Tervezés automatizálásához.
PROJECTEXPERT	Projectépítés tervezési fázisához.
ZEXPERT	Banki alkalmazásokhoz.

Forrás: [1], [12], [15], [17]

## II.2. A Prolog nyelv jellemzőinek áttekintése

A programozási nyelvek csoportosíthatóak az alábbiak szerint:

- *Professzionális* ill. *amatőr* nyelvek,
- *Alacsony szintű* ill. *magasszintű* nyelvek,
- Számítási modellek alapján:
  - *Neumann-elvű*,
  - *automata-elvű*,
  - *funkcionális*,
  - *logikai* nyelvek. [19]

A programozási nyelveket programozási paradigmák szerint az alábbi módon csoportosíthatjuk:

- *Imperatív* nyelvek
  - Procedurális – eljárás-orientált
  - Objektum-orientált
- *Deklaratív* nyelvek
  - Funkcionális,
  - Logikai (szabály alapú). [20]

A programozási paradigma alatt olyan mintákat, módszereket, technikákat értünk, amilyen módon a programozási nyelvet használni tudjuk.

Az imperatív programnyelvek *algoritmus-vezérelt* nyelvek. Működésük a *Neumann*-architektúrához köthető. Használatuk során a programozó mondja meg, hogy *mit, hogyan* kell *megoldani*. A programok, melyek további önálló részekre bonthatók, *utasításokból* állnak. A programok az alapvető vezérlési szerkezetek felhasználását támogatják (szekvencia, elágazás, ciklus). Ezekben a nyelvekben létezik az *értékkadás* művelete, használhatók *változók*.

Alcsoportjai:

- Eljárásorientált nyelvek, pl.: C, Ada, Pascal, Modula-2, Basic, Fortran, Cobol
- Objektum-orientált nyelvek, pl.: C++, C#, Java, Delphi, Simula

A deklaratív programnyelvek *nem-algoritmikus* nyelvek. Kevésbé, illetve *nem* építenek a *Neumann*-architektúrára. Használatuk során a programozó csak azt mondja meg, hogy *mit* kell megcsinálni (*nem* mondja meg, hogy *ezt hogyan* kell végrehajtani). További jellemzőik még, hogy nincs utasítássorozat, nincs értékkadás, nincsenek változók.

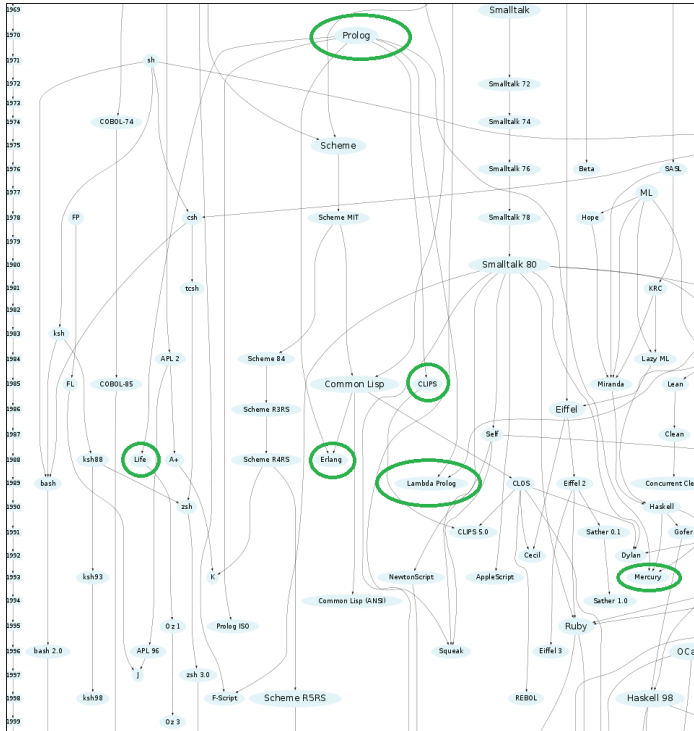
Alcsoportjai:

- Funkcionális nyelvek, pl.: Lisp, Miranda
- Logikai nyelvek, pl.: Prolog

A **Prolog** nyelv deklaratív, logikai, nagyon magas szintű programozási nyelv, az ember gondolkodásához áll közel. A Prolog nyelvből fejlődtek ki a következő programnyelvek is: Clips, Erlang, Mercury, Lambda Prolog, Life, melyet az alábbi ábra szemléltet.

## 2. ábra

A Prolog nyelv helye a nyelvek családjában



Forrás: [21]

A **Prolog** nyelvet az 1970-es évek elején fejlesztették ki Alain Colmerauer vezetésével. A nyelv neve a **P**rogrammation en **l**ogique (logikai programozás) rövidítéséből ered. A Prolog nyelv az egyik legfontosabb eszköze a mesterséges intelligenciát alkalmazó programozásnak, valamint a szakértői rendszereknek. Sok, jelentős szakértői rendszert fejlesztettek Prolog-ban. [22]



Szeredi P. és Benkő T. [20] így fogalmaz a Prolog nyelv fejlesztésének legintenzívebb korszakáról: „1981-ben a japán kormány egy nagyszabású számítástechnikai fejlesztési munkát indított el, az ún. „ötödik generációs számítógéprendszerek” projektet, amelynek alapjául a logikai programozást választották. Ez nagy lökést adott a terület kutató-fejlesztő munkáinak, és megjelentek a kereskedelmi Prolog megvalósítások is.”

A Prolog nyelv egy magyar nyelvű változatát (MProlog – moduláris Prolog) Szeredi Péter fejlesztette ki.

Márkus Zs. [23] alapján a Prolog nyelv szintaktikai szabályai, valamint működési mechanizmusa így foglalható össze:

- A Prolog program mondatok sorozatából áll. A mondatok végén pont található.
- A Prolog mondatokat más néven állításoknak nevezhetjük, oly módon, hogy minden mondat egy logikai állítás, amelynek igazságértéke van. Fontos megjegyezni, hogy egy Prolog állítást mindig igaznak tekintünk.
- A :- szimbólum a logikai IMPLIKÁCIÓ ( $\rightarrow$ ), a vessző ( , ) a logikai ÉS míg a pontos vessző ( ; ) a logikai VAGY jele.
- A nagybetűvel írt azonosítók változót, a kisbetűvel írottak konstans jelölnek.
- Egy elemi állítás egy predikátum névből és argumentum(ok)ból áll. Minden argumentum egy kifejezés. Ha nincs argumentum, akkor a predikátum nulla argumentumú, egyébként egyargumentumú, kétargumentumú, stb. A kifejezés az vagy egy konstans, vagy egy változó, vagy egy összetett kifejezés lehet.
- Az implikáció formájú állítások a **szabályok**. A szabály bal oldala a szabály feje, míg a jobb oldala a szabály törzse. A szabály feje egy elemi állítást, míg a szabály törzse több elemi állítást is tartalmazhat vesszőkkel (és-sel) elválasztva. Pl.:

**vesznek\_autót(X):- robinak\_tetszik(X), jutkának\_tetszik(X), van\_rá\_pénz(X).**

- A feltétel nélküli állítások a **tényállítások**.
- Egy Prolog program csak egy leszűkített, a szabályokkal és tényállításokkal definiált világban működik. Azokat az ismereteket veszi figyelembe, amelyeket definiáltunk.

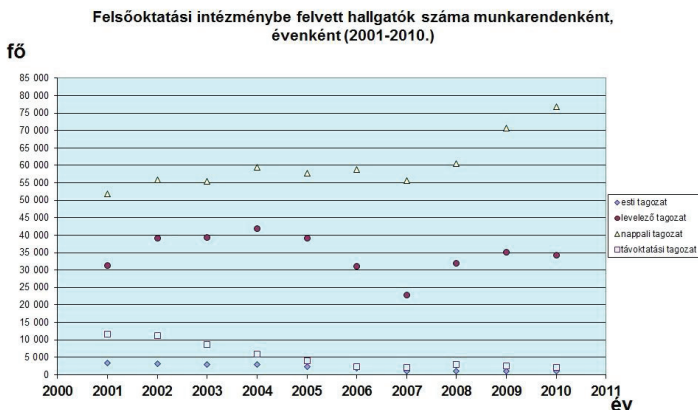
Egy Prolog program futtatásához szükség van egy feladatmegoldó algoritmusra is (következtetési rendszer). A Prolog beépített, saját következtetési rendszerrel, feladatmegoldó stratégiával rendelkezik. Egy feladat megoldását a célállítás - mely egy vagy több elemi állítás vesszővel elválasztva - végrehajtása jelenti. [23]

A Prolog nyelvben két fontos, beépített mechanizmus működik, a ***mintaillesztés*** és a ***visszalépés (backtrack)***. A mintaillesztés két kifejezés közötti, argumentumonkénti összehasonlítást jelent. Ha egy feladat egyik részfeladatát nem sikerül megoldani, akkor a teljes feladat sem oldható meg. Ha egy részfeladatot nem sikerült megoldani, akkor abban az esetben, ha ezt a részfeladatot meg lehet oldani másképpen is, visszalép a feladatmegoldó algoritmus. Amennyiben nem lehet megoldani másképpen, akkor a program leáll. [22], [23]

### III. Pedagógiai kihívások az Internet világában

A magyar felsőoktatásban az elmúlt években a távoktatási forma egyre inkább háttérbe szorult mind arányaiban (2001-ben a felvett hallgatók 11,8 %-a, míg 2010-ben 1,8 %-a távoktatási tagozatra nyert felvételt), mind pedig létszámát tekintve (2001-ben 11 553 fő, 2010-ben 2 012 fő a távoktatási tagozatra felvett hallgatók száma), melyet a 3. és 4. ábra szemléltet [24]. Ennek számos oka van, pl. a korábban a felsőoktatásból kimaradt felnőttek tömeges jelentkezési hulláma véget ért, a kétszintű képzésre való átállás átmeneti időszaka az idősebb korosztályt érzékenyebben érintette, valamint évről évre csökken a költségtérítéssel képzést választók aránya (mivel a többség államilag támogatott képzésben szeretne részt venni). Emellett azonban úgy gondolom, helytálló érv lehet az is, hogy a távoktatás korábbi rendszere, elsősorban módszerei, eszközei vitatható hatékonyságának köszönhetően, nem tudott átütően, tartósan teret hódítani felsőoktatásunkban.

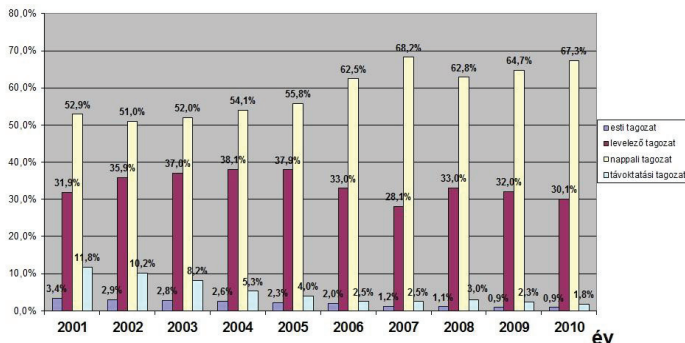
#### 3. ábra



Forrás: [24] alapján saját szerkesztés

#### 4. ábra

Felsőoktatási intézménybe felvett hallgatók megoszlása munkarendenként, évenként (2001-2010.)



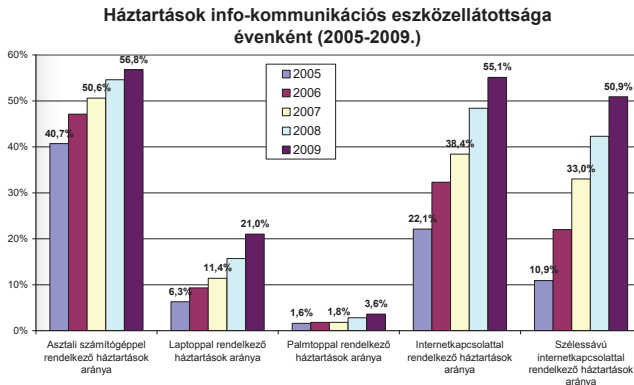
Forrás: [24] alapján saját szerkesztés

### III.1. A lakosság info-kommunikációs eszközellátottsága

Az info-kommunikációs eszközök, valamint az Internet térhódítása már Magyarországon is gyökeresen alakítja át a mindennapokat, az elektronikus vásárlástól, banki ügyintézésen át a hivatalos ügyek elektronikus ügyintézéséig, így természetesen az oktatásra is komoly hatással van. A Központi Statisztikai Hivatal adatai alapján néhány fontos adatra hívom fel a figyelmet. A háztartások info-kommunikációs eszközellátottságát, illetve használatát szemlélteti az 5. és a 6. ábra.

Fontos kiemelni az 5. ábráról, hogy 2009-ben az asztali számítógéppel rendelkező háztartások aránya 56,8%, az internetkapcsolattal rendelkező háztartások aránya 55,1% és az e-learning sikeréhez fontos, szélessávú internetkapcsolattal rendelkező háztartások aránya 50,9%. Mivel ezek az adatok a lakosság teljes összetételére vonatkoznak, ezért joggal feltételezhető, hogy az oktatási rendszerünkben lévő (és leendő) hallgatók háztartásait figyelembe véve ezek az arányok még magasabbak. Az ábrán nem került feltüntetésre a háztartások mobiltelefonnal való ellátottsága, ugyanis 2009-re a mobiltelefonnal rendelkező háztartások aránya 90,4 %. [25]

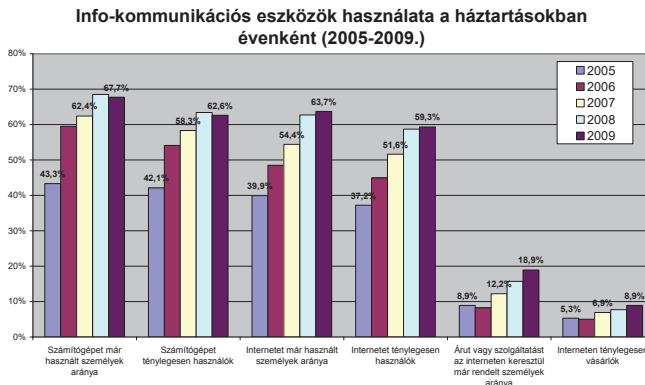
5. ábra



Forrás: [25] alapján saját szerkesztés

A 6. ábra az info-kommunikációs eszközök használatát részletezi. Fontos kiemelni, hogy mind a számítógépet, mind az internetet használók aránya az évek során növekszik, valamint nő azok aránya is akik az internetet vásárlásra, vagy egyéb szolgáltatás igénybevételére használják. A felmérés során a tényleges használók csoportjába azokat a személyeket sorolták, akik az utóbbi 3 hónap során használták az adott eszközt, illetve szolgáltatást.

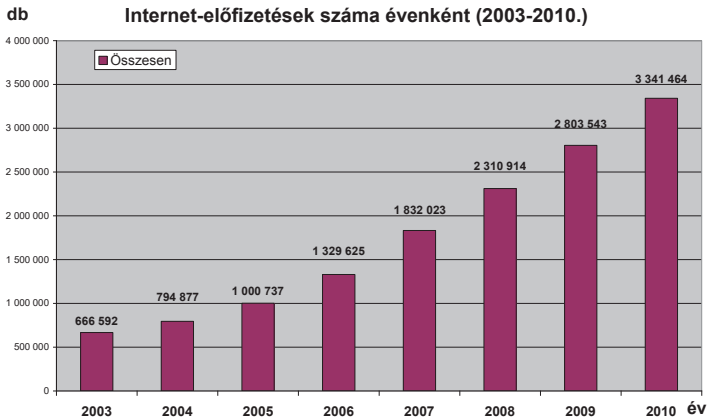
6. ábra



Forrás: [25] alapján saját szerkesztés

A 7. ábra megmutatja, hogy összesen mennyi az internet-előfizetések száma az adott évben. Jól látható az ábrán, hogy az internet-előfizetések száma 7 év alatt ötszörösére növekedett (2003-ban 666 592, míg 2010-ben 3 341 464 az előfizetések száma) [26].

**7. ábra**



Forrás: [26] alapján saját szerkesztés

Összegzőképpen megállapítható tehát, hogy a mai Magyarországon széles körűen rendelkezésre áll az a technológia, amely az e-learning sikeréhez nélkülözhetetlen.

Az a paradox helyzet alakult ki mára, hogy pont akkorra csökken le a hallgatók jelentkezési száma, felvétele a távoktatási formákra, amikor a legegyszerűbb lenne kivitelezni ezt a képzési formát. Véleményem szerint, az e-learning segítségével, a magyar felsőoktatásban újra a távoktatási formák bővülését fogjuk tapasztalni.

## **III.2. Programozott oktatás használata a távoktatásban e-learning segítségével**

Az előző részben láthattuk, hogy az info-kommunikációs eszközök tömeges elterjedése, valamint az Internet térhódítása milyen komoly mértékű már Magyarországon, mely természetesen az oktatásra is jelentős hatással van.

Ebben a fejezetben bemutatom a távoktatás, a programozott oktatás és az e-learning alapvető jellemzőit, ismertetem a Moodle legfontosabb tudnivalóit, valamint megoldási javaslatokat teszek e-learning kurzusok megvalósítására.

### **III.2.1. Távoktatás és programozott oktatás**

Csoma Gy. [27] szerint a távoktatás a tanulás távirányításának egyik formája. *„Előírt és tananyagga rendezett ismeretek, gondolkodási és – korlátozottan – cselekvési műveletek elsajátíttatására irányul, meghatározott követelmények teljesítése, megtervezett tudásszintek elérése érdekében. Arra törekszik, hogy a tanulási folyamat minden mozzanatát a kezében tartsa, ideértve a jártasságok, készségek kifejlesztését is. Ezért a tanulást folyamatosan, lépésről lépésre irányítja, rendszerezi és átfogóan szervezi.”*

Kokovay Á. [28] így foglalja össze a távoktatás főbb elemeit:

- *„a távoktatásban az oktató és az oktatott személyes kapcsolata jelentős mértékben módosul, egyes esetekben akár meg is szűnik,*
- *a tanulási folyamat irányított, vezérelt,*
- *a tanulás irányításában a személyes kapcsolatok helyét nagyrészt a médiumok együttese veszi át,*
- *a tanuló rendszeresen válogatott és összeállított tananyagot kap,*
- *kiemelkedő jelentősége van a motiválásnak,*
- *a tanulási folyamatban a tanuló önállósága növekszik,*
- *a tanulás nagy mértékben alkalmazkodik a tanuló képességeihez, körülményeihez.”*

A távoktatásban jelentős szerepe van a távirányításnak. A tanulás közvetett irányítására, a tanulók egyéni tanulására az egyik legjobb módszer a programozott oktatás. A megfelelő programozott tananyagok ugyanis nagyon hasznos visszacsatolást is tartalmaznak a tanulás folyamán.

A. Vincent [29] szerint a programozott oktatási anyagok készítése ugyan fáradtságos munka, de mindenképpen kárpótolja a készítőt az a fajta érdeklődés és tanulási kedv, amely megnyilvánul ennek hatására a tanulók részéről.

L.N.Landa [30] azt javasolja, hogy a bonyolult műveleteket kisebb, elemi műveletekre tagoljuk mindaddig, amíg olyan elemi műveletekhez nem jutunk, amelyeket mindig mindenki egyértelműen hajt végre.

Tóthné Köröspataki Kiss Á. [31] megfogalmazása szerint: *„A programozott oktatás olyan individualizált oktatás, amely a tanulókat előre megtervezett lépések (programlépések) sorozatán tananyagelemekből felépített algoritmus szerint vezeti a kitűzött tanulási cél eléréséhez. A programok biztosítják a tanulásban az egyéni haladási ütemet, illetve a programozás jellegének megfelelően az önellenőrzést.”*

Nagy előnye, hogy a tanulási nehézségekkel küszködőknek az egyéni próbálkozás lehetősége biztonságot nyújt, mert nem válik nyilvánossá a hibázás a tanulótársak előtt [31].

A programozott oktatás legnagyobb hulláma a 20.század közepére tehető. A nyugati világban a háborús tömeges kiképzésekkel kapcsolatosan jelentkezett. Magyarországra a 60-as években érkezett, a 70-es évekbe tehető komolyabb elterjedése [32]. Az Egyesült Államokban a tömeges oktatási feladatokat oldották meg programozott oktatással. Az amerikai programozott oktatás számos európai országban honosításra került és jelentős volt a hatása az angliai Open University létrejöttére is [33].



A programozott oktatás előnyei:

- jól felépített oktatási tematika, oktatói program,
- sok, közvetlen visszajelzés a tanulónak, önellenőrzés, visszacsatolás,
- erőteljesen az egyéni munkára épít, aktív közreműködést igényel a tanulótól,
- egyéni haladási ütem, képességeknek megfelelő elágazási pontok,
- gazdaságos, mivel kevés tanerőt igényel nagy tanulói tömegekhez.

A programozott oktatás hátrányai:

- a pedagógusi közreműködés minimalizálása,
- egyirányúság,
- a számonkérés (önellenőrzés esetén) az emberi nyelv kezelésére korlátozottan képes (csak egy előre megadott válaszalmazból lehet választani),
- gondolati merevség,
- elnyomhatja a tanulók intellektuális aktivitását, korlátozottan alkalmas ugyanis az alternatív gondolkodás fejlesztésére,
- leszoktathatja a tanulót a kreatív gondolkodásról.

A programozott oktatási formák előnyeit, hátrányait megvizsgálva kijelenthető, hogy a programozott oktatás nagyon alkalmas a bonyolultabb ismeretek egyszerű lépésekben való feldolgozására, az egyszerű feladatok (rutinfeladatok) betanítására, az elsajátított ismeretek számonkérésére. Kifejezetten hatékony lehet a hiányos előképzettséggel rendelkező hallgatók szintrehozásában. Ezek figyelembevételével úgy gondolom, hogy használata felsőoktatásunkban, különösen a távoktatásban, elengedhetetlen.

Ratkó I. [34] felveti annak a lehetőségét, hogy az Internet széleskörű felhasználásával a programozott oktatás az Interneten is kerüljön megvalósításra.

### III.2.2. E-learning rendszerek jellemzői

Komenczi B. [35] szerint „*az e-learning a számítógép és a hálózati adatbázisok, illetve internetes kommunikáció használatával, a tanulási folyamat egészének rendszerszemléletű megközelítésével, illetve hatékony rendszerbe szervezésével törekszik a tanulás eredményességének javítására.*”

R. C. Clark és R. E. Mayer [36] szerint az e-learning egy olyan számítógépes alkalmazás, amely:

- összetett oktatási módszereket tartalmaz, melyek közös célja, hogy segítse a tanulás folyamatát egyszerű példákon és összetett feladatokon keresztül,
- a tanulás sajátosságainak megfelelő tartalmakat kezel,
- különféle média-elemeket (kép, hang, video) használ, melyek célja az ismeretanyag átadása,
- olyan új tudást és képességeket épít, amely által jobb teljesítményt ér el a tanuló.

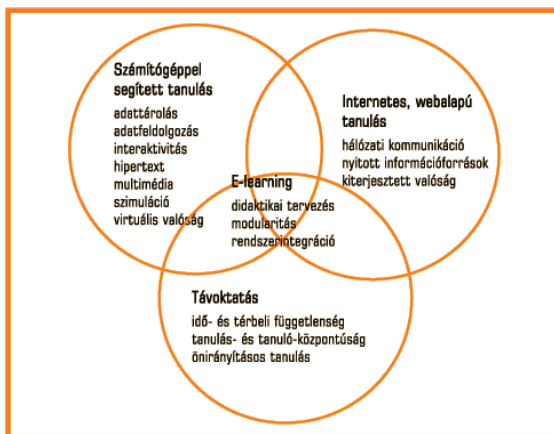
Kokovay Á. [28] így fogalmaz:

„*A távoktatási rendszerekhez képest az e-learning két alapvető új eleme, hogy*

- *a tananyag elektronikus formában jut el a hallgatóhoz,*
- *az átvitel közegei az internet, illetve intranet hálózatok.*”

Az e-learning kapcsolatát a távoktatással, számítógéppel segített tanulással, valamint a web-alapú tanulással mutatja a 8. ábra.

## Az e-learning összetevői



Forrás: [37]

Ebből jól látható, hogy az e-learning nem azonosítható sem a számítógéppel segített tanulóval, sem az internetes, webalapú tanulóval, sem pedig a távoktatással. Az e-learning tehát egy új oktatási forma, mely egy új minőséget jelent.

Az e-learning legfontosabb előnyei M. J. Rosenberg [38] nyomán így összegezhetők:

- a tananyag-tartalom könnyen testreszabható, bármikor könnyű frissíteni az információkat,
- a hozzáférhetőség egyszerű, a nap 24 órájában elérhető,
- nagyon jelentős a közösségépítő jellege, mely hatalmas motiváló erő,
- a felhasználók növekvő létszáma esetén rugalmasan, könnyen átméretezhető,
- jelentős eredmény-javulást lehet elérni a felhasználók teljesítményében,
- gazdaságos (a kialakításának kezdeti magas költségei ellenére).

Az e-learning hátrányai között a legmeghatározóbb, hogy csak egy adott információkommunikációs infrastruktúra esetén kivitelezhető, tehát a felhasználói oldalon a számítógépek hiánya, valamint a megfelelő (szélessávú) internet-elérés elégtelensége nem teszi lehetővé az e-learning használatát. Természetesen ezen eszközöknek a megfelelő kezelésével (digitális írástudás) is rendelkeznie kell a tanulóknak a sikeres használathoz.

A hatékony e-learning megvalósítását napjainkban már sokfajta eszköz, technika és technológia segíti. Az e-learning egyik legdinamikusabban fejlődő eszköze a Moodle, amely egy nyílt forráskódú, ingyenes szoftver. A Moodle egy mozaikszo, a **Modular Object-Oriented Dynamic Learning Environment** (Moduláris Objektum-Orientált Dinamikus Tanulási Környezet) kifejezés rövidítése. A Moodle futtatásához szükséges szoftverkönyezetet a szerver oldalon: Apache webszerver, PHP szkriptnyelv és MySQL adatbázisszerver, valamint ezek futtatására alkalmas operációs rendszer (pl.: Unix/Linux, Windows, MAC OS X), míg a felhasználói oldalon egy tetszőleges web-böngésző program (pl.: Mozilla Firefox) jelenti [39]. Annak ellenére, hogy a szoftvert csupán 1999-ben jegyezték be, 2008-ra már több mint 70 nyelvre fordították le és a világ 196 országában használják [40].

A Moodle alapvető szolgáltatásai közé tartozik a tananyagfejlesztés, kurzusszervezés, valamint a kommunikáció.

A Moodle legfontosabb szolgáltatásai:

- tartalomkezelés (SCORM tananyagok, állományok kezelése pl.: saját készítésű oktató videók, videomegosztókon található online videók, online logikái, szimulációs játékok, stb.),
- kommunikációs lehetőségek (chat, fórum, elektronikus levelezés, skype),
- ellenőrzés és értékelés módjai (leckék, tesztek, fogalomtár). [41]

A Moodle nemcsak a távoktatásban használható fel, hanem az egyéb (nappali, levelező, esti) képzési formákban is. Ekkor azonban már inkább blended learning-ről lehet beszélni e-learning helyett.

A blended-learning egy olyan vegyes oktatási forma, amely a hagyományos oktatási formák mellett az online tanulási technikákat is alkalmazza. Ez a kombinált oktatási forma hatékonyan ötvözi mindkét változat előnyeit [42].

A fentiek figyelembevételével elmondható, hogy az e-learning és így a Moodle megfelelő eszköze a távoktatásos oktatási formának. A Moodle könnyen kezelhető, gazdaságos, és magában foglalja mindazokat a lehetőségeket, amelyek lehetővé teszik a különféle oktatási módszertanok (és így a programozott oktatás) megvalósítását. Nagy előnye, hogy a módszerek vegyes alkalmazását jól támogatja, így egyik módszer sem lesz kizárólagos az oktatás során. Ezáltal pedig csupán az intézmény, illetve a tanár kreativitására van bízva, hogy hogyan tud élni a különféle módszerek előnyeivel, s hogyan tudja kiküszöbölni az egyes módszerek hibáit.

### **III.3. Megoldási javaslatok**

Ebben a részben összefoglalom azokat a kritériumokat, módszertani megoldási javaslatokat, amelyeket elengedhetetlennek tartok a hatékony és népszerű távoktatáshoz. Ezek a megállapítások véleményem szerint receptként is szolgálhatnak egy ilyen kurzus megszervezéséhez.

Nagyon fontosnak tartom már a kurzus megtervezésének elején az oktatási célkitűzések pontos megfogalmazását, a tananyag tematikájának rögzítését, a tananyag feldolgozásához szükséges előismeretek megadását. Fontos a kötelező tananyagot elérhetővé tenni (mely összefoglalja az adott téma tudását elektronikus és/vagy nyomtatott formátumban). Célszerű a minimum követelmények rögzítése is (pl. fogalomtárban a minimum fogalmak, a vizsgára való beugró tesztek elérhetővé tétele, minták megadása kiselőadások, beadandó dolgozatok, programok követelményeihez, stb.).

A tananyag törzsanyagának feldolgozása programozott módszertant követő leckeekben történjen, mely biztosítja az egyéni képességeknek megfelelő haladási ütemet.

Nagyon nagy előnye a Moodle-nek, hogy ezek a leckék nemcsak egyszerűen elkészíthetőek, hanem könnyen módosíthatóak is (nem merev), ha valami olyan probléma van, amely sok tanulónak okoz gondot, pl. tömegesen rotnak el bizonyos kérdéseket, az nagyon egyszerűen korrigálható. Ez egy drága oktató CD-nél, vagy korábban egy nagy példányszámban kiadott programozott tankönyv esetén megoldhatatlan volt. Nagyon fontos, hogy a kommunikációs lehetőségek (e-mail, chat, skype, msn, fórum) egységes keretben, lehetőleg egy helyen legyenek integrálva. Érdeemes biztosítani konzultációs lehetőségeket az internet segítségével (skype, msn) is, illetve személyesen is, megadott időpontokban.

A kiscsoportos foglalkozások nem jelentik szükségszerűen csak a tanárral való kapcsolatot, találkozást. Mivel a kurzusban résztvevők elérhetik egymást, fel tudják venni egymással is a kapcsolatot. Lehet, hogy nem is laknak messze egymástól a hallgatók, így könnyen kapcsolatba tudnak lépni egymással személyesen is, tehát nemcsak a virtuális térben jöhet létre egy csoport, kisközösség. Ez a fajta kapcsolat azért is elengedhetetlen, mivel sokszor tapasztaltam hallgatóink körében azt, hogy aki lemaradt a csoporttól, magára maradt, úgy érezte egyedül küzd az „elemekkel”. Ezeknek a tanulói csoportoknak (akár virtuális, akár személyes) a létrejöttét segíteni, ösztönözni kell.

A programozott oktatás alkalmas nagy létszámú hallgatói csoportok oktatásának gazdaságos megvalósítására, viszont a hátrányait is ismerni kell, mivel ez sem kizárólagos módszer. Hátrányai kiküszöbölésében azonban segít a Moodle, mely hatékonyan támogatja a konstruktivista pedagógiát.

Bessenyei I. és Tóth Zs. [43] megfogalmazása szerint: *„A konstruktivista tanulásméлет szerint a tanulók úgy építik fel a tudásukat, hogy korábbi tapasztalataik, ismereteik, élményeik szerint formálják az új ismereteket. A konstruktivista felfogásban a tanuló nem üres „edény”, amit meg kell tölteni a már létező tudással, hanem aktívan részt vesz a tudás felépítésében. Ha az iskolai oktatás képes a tudásépítés összetett, szerteágazó és egyénenként különböző hangsúlyokkal bíró jellegét megragadni és leképezni, sokkal eredményesebb lehet, mint a korábbi merev struktúrák.”*

A Moodle számos, konstruktivista pedagógiát támogató eszköze közül ki kell emelni a fórum, wiki, fogalomtár, adatbázis, üzenetküldés lehetőségeket, melyek segítségével hatékony lehet a tudásmegosztás, valamint a tapasztalatcsere. Emellett nagyon egyszerűen, könnyen ki lehet használni a Moodle multimédiás szolgáltatásait is. Így amit lehet, érdemes szemléltetni pl.: saját készítésű videókkal, gazdag illusztrációkkal, hasznos diagramokkal, interneten elérhető tartalmakkal, online játékokkal stb. Oda kell figyelnie a tanárnak, hogy a gyorsan, jól teljesítő tanulóknak is tudjon újat mutatni, ne szoktassa le a tanulót a kreatív gondolkodásról, közös munkákat is kezdeményezzen.

A konstruktivista pedagógia, valamint a multimédiás lehetőségek gazdag választéka lehetővé teszi a programozott oktatás hátrányaiként felsorolt jellemzők kiküszöbölését is. A Moodle-ben ugyanis lehetőség van (és egyszersmind követelmény is) a pedagógus aktív közreműködésére, a tudásmegosztás és tapasztalatcsere lehetőségei pedig fokozzák a tanulók intellektuális aktivitását, kreatív gondolkodást igényelnek (ezáltal a gondolati merevség, és az egyirányúság is jelentősen lecsökken a tanítási-tanulási folyamatban).

Nagyon fontos a tesztek megoldásának követése: a tanuló nemcsak „magának” tölti ki a tesztet, hanem a tanár is látja, hogy milyen eredménnyel oldotta meg azt, foglalkozott-e a leckével. (Ezek a próbálkozások a többi tanuló elől rejtve maradnak, csak a tanárnak van jogosultsága ezek megtekintésére.) Fontos, hogy a hallgatók ne érezzék egyedül magukat a képzésben, mindig tudják, hogy tevékenységüket szemmel követi a tanár, tudja, hogy ki, mikor, mit hajtott végre, vagy mit nem (ez hat bátorítólag is, de természetesen számonkérőleg is).

A leckék, tesztek értelmes, hatékony felhasználása elengedhetetlen ennek a módszernek a sikeréhez, ahogyan azt Bakó M. [44] is kifejti: *„A diákok hozzáállását jelentősen megváltoztatja, ha ott lebeg felettük Damoklész kardja, tehát ha nem csak játszadoznak vakvilágba a tesztekkel, hanem a tanár minden egyes megoldásról értesül.”*

Ügyelni kell arra, hogy a számonkérés (vizsga) ne csak a tesztek kitöltéséből álljon. A tesztek kitöltése és jó megoldása nem elegendő az adott témában való kellő jártasság megszerzésében, az elsajátított ismeretekről szóban, vagy írásban (programozáskor gyakorlatban) is számot kell tudniuk adni. A leckék során felmerülő kérdéseket úgy is meg lehet oldatni, hogy papírra írják a megoldást, s a tanuló egyénileg ellenőrzi magát, hogy jól válaszolt-e. Ez a módszer tehát elmozdul abból a kritikus pozícióból, hogy csak tesztkérdéseket lehet feltenni, s annak megfelelően lehet továbbléptetni egy leckében, érdemes a hallgatót a saját maga szigorú ellenőrzésére is rászoktatni (önellenőrzés megtanítása).

Ezen adatok, ismeretek, jellemzők alapján fogalmaztam meg tézisemet, miszerint:

**A távoktatás hatékonysága e-learning rendszerben, a konstruktivista pedagógia, valamint a programozott oktatás módszereivel jelentősen növelhető.**



#### **IV. Intelligens segédanyag az anyagszerkezet oktatásához**

Ma már sajnos szomorú tény, hogy a természettudományos tárgyak népszerűtlenebbek mind az általános iskolákban, mind pedig a középfokú oktatási intézményekben. Véleményem szerint ezen tárgyak népszerűségét hatékonyan lehet növelni olyan intelligens oktatási segédanyagok elkészítésével, amelyek segítségével a természeti törvények könnyebben tanulhatókká válhatnak.

A feladatok tudásbázisának adott formába öntése (tudásreprezentáció) segíti az ismeretek pontos rögzítését, a lényeg megragadását, a fontos és nem fontos információk elkülönítését. A feladatok megoldása saját kitalálási stratégiát, számos esetben eredeti ötletet követel meg.

Ebben a fejezetben bemutatok egy olyan intelligens oktatási segédanyagot, amelyet Prolog nyelven fejlesztettem, s amelynek sikeréből, oktatási tapasztalataiból fogalmaztam meg a következő tézisemet, miszerint:

**A természeti törvények szabályszerűségeinek oktatásakor a szabályalapú szakértői rendszerek készítése indokolt, különösen a tanulókkal közös munkával.**

Számítástechnika tanári diplomám mellett okleveles vegyész és kémia tanár diplomával is rendelkezem. A tanulmányaim során mindig nagyon foglalkoztatott az anyag szerkezete. Tanárként azt tapasztaltam, hogy az anyag szerkezetének tanítása, illetve a tanulók részéről ennek a tananyagrésznek az elsajátítása sok nehézségbe ütközik. Nagyon sok szabályt kell megtanulnia a diáknak, hogy megértse az elektronszerkezet felépítését. Az anyagok viselkedésének a megértéséhez azonban nélkülözhetetlenek az anyagszerkezeti alapok, így ezekre az ismeretekre épül a teljes további feldolgozandó tananyag.

Kutatásaim során arra a felismerésre jutottam, hogy a megvalósított és sikeres szakértői rendszerek közül nagyon sok található a kémia területéről.

Olyan alkalmazásokban lehet sikeresen használni ezeket, ahol sok, jól reprezentálható ismeret áll rendelkezésre. A következtetések levonásához ezt a sok szabályt kell egyszerre figyelembe venni, alkalmazni. Ezen szakértői rendszerek közül több is Prolog nyelven került megvalósításra.

Nagyon sok hasznos szakértői rendszer létezik pl. a Dendral, MetabolExpert, Mycin, Internist, Guidon, MProlog Shell, Hearsey, stb. A Dendral-t, amely az első sikeres tudásalapú rendszer, 1979-ben alkották. A Dendral (Dendritic Algorithm) ismeretlen szerves vegyületek molekulaszervezetének meghatározására alkalmas tömegspektrográfiai és mágneses rezonancia mérési adatokból. A NASA Mars programjában az egyik műhold számára készült. [12]

Külön kiemelném azt a tényt, hogy az első sikeres szakértői rendszer **kémiai** tudást használó szakértői rendszer volt. További híres kémiai szakértői rendszerek a MetabolExpert, amely kémiai, orvosi, biológiai előrejelzésekhez használható, valamint a PIRExS, amely szervetlen kémiai reakciók előrejelzésére alkalmas.

Ezen előzmények után úgy gondoltam, hogy indokolt lehet az anyagszerkezeti alapok oktatásához szakértői rendszert tervezni, illetve megvalósítani.

## IV.1. Az intelligens tananyag

A kémiai feladatok, problémák megoldása szabályalapú rendszerek segítségével nagyon hasznosnak bizonyul. Érdekességgént említhető a periódusos rendszer felfedezése is. A XIX. században D. I. Mengyelejev a kémiai elemeket növekvő atomtömeg szerint állította sorba. Mivel akkoriban még ismeretlen volt az atom felépítése, az atomok legjellemzőbb adatának az atomtömeget tekintették. Azt vette észre, hogy minden nyolcadik elem hasonlít egymásra. Ezeket az elemeket egymás alá írta, melynek során oszlopokat kapott. Ezt a szabályszerűséget követve egy csodálatos természeti törvényt sikerült felfedeznie.

Egyes helyeket üresen hagyott táblázatában, és meg volt róla győződve, hogy ezeket az elemeket fel fogják fedezni.

Nem tudta azonban az egy oszlopba írt elemek kémiai hasonlóságát megmagyarázni, ugyanis nem ismerte az atomok szerkezetét. Következtetési azonban kiállták az idő próbáját: 1869-ben megalkotott rendszerét (melyet nyugodtan nevezhetünk akár szabályalapú rendszernek is) 60 évvel később igazolták. Az elemek sorrendjét ugyanis nem az atomtömeg, hanem a protonok (és semleges atomok esetén az elektronok) száma határozza meg. Az atomok kémiai tulajdonságai pedig az elektronburok sajátosságaitól függenek. [45]

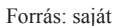
Ebben a fejezetben egy intelligens tananyagot mutatok be, melyet az anyagszerkezet oktatásában nagyon jól lehet hasznosítani. Az elkészült intelligens segédanyag az elemek tulajdonságait (elektronszerkezet, oxidációs számok, elektronegativitás) rendszerezi, s alkot ezen szabályokból és tényekből egy szabályalapú rendszert. Mintegy 50 szabályt és 150 tényállítást tartalmaz, de fejlesztése nem ért véget. A cél az volt, hogy a tudásanyag úgy legyen ábrázolva, hogy egyrészt megoldja a kitűzött feladatokat, másrészt a tanulók számára is könnyen megérthető, illetve elkészíthető legyen. Véleményem szerint a legnagyobb erénye abban rejlik, hogy arra ösztönzi a tanulókat, hogy ők is készítsenek ilyen szabályalapú rendszereket. Vagyis próbálják meg a tudásanyagot megfelelő formába önteni, majd értelmes kérdések után hasznos következtetésekre jutni. Ennek segítségével pedig az egyik legfontosabb oktatási célt lehet elérni, vagyis a problémamegoldó képességük válik fejleszthetővé a tanulóknak.

Az intelligens tananyag Win-Prolog programmal készült. A program elkészítésében felhasználásra kerültek a [23], [46] és [47] forrásmunkák.

Az intelligens tananyag forráskódja az *A függelékben* található, melyből a legfontosabb részletek ezen a helyen is kiemelésre kerültek.

A 9. ábrán a tananyag főablaka látható.

### *Az intelligens tananyag főablaka*



10. ábra

### Periódusos rendszer

Forrás: saját

## IV.2. Anyagszerkezeti alapok a feladatmegoldásokhoz

Az **atom** pozitív töltésű **atommagból** és körülötte mozgó negatív töltésű **elektronokból** áll. Az atommag pozitív töltésű protonokból és semleges töltésű neutronokból áll. A semleges atomban a pozitív töltések száma megegyezik a negatív töltések számával. Az atom rendszáma egyenlő a protonok (és semleges atom esetén az elektronok) számával. Az atommag körül lévő elektronok összességét **elektronburoknak** nevezzük. Az elektronok rétegesen, héjakat alkotva építik fel az elektronburokot. [45]

Az elektronok állapotának a jellemzésére összesen 4féle kvantumszámot ( $n$ ,  $l$ ,  $m_l$ ,  $m_s$ ) használunk. Az  **$n$**  a **főkvantumszám**, az  **$l$**  a **mellékkvantumszám**, az  **$m_l$**  a **mágneses kvantumszám**, az  **$m_s$**  pedig a **mágneses spinkvantumszám**. [45]

A **főkvantumszám** értékei alapállapotban **1, 2, 3, 4, 5, 6 és 7** lehet. Az azonos főkvantumszámú elektronok „**elektronhéjat**” alkotnak. A héjaknak nevet is adtak, az előző sorrend alapján **K, L, M, N, O, P, Q** héjról beszélhetünk.

A főkvantumszám a mellékkvantumszámmal együtt az elektron összes energiáját határozza meg. A **mellékkvantumszám** az a szám, amely kifejezi a különböző energiaszinteket egy főkvantumszámon belül, értékei **0-tól ( $n-1$ )-ig** terjedhetnek. A mellékkvantumszámokat betűkkel is jelölhetjük: 0 – s, 1 – p, 2 – d, 3 – f. A héjakon az azonos mellékkvantumszámú pályák **alhéjakat** hoznak létre. Azok a csoportok tartoznak egy mezőbe (s-mező, p-mező, d-mező, f-mező), amelyekben ugyanaz az alhéj kiépülése történik. [48]

A **mágneses kvantumszám** megadja a mellékkvantumszám által megszabott pályáformák térbeli irányát. Értéke  **$-l$ -től  $+l$ -ig** bármely egész szám és 0 lehet. Az azonos mellékkvantumszámokhoz ( $l$ ) tartozó **pályák száma** összesen  **$2 \cdot l + 1$**  lehet. Az  **$n$** . héjon a pályák maximális száma  **$n^2$**  ( $n$  a főkvantumszám).

A **mágneses spinkvantumszám** értéke  **$+1/2$  illetve  $-1/2$**  lehet, mely az elektron két lehetséges állapotát fejezi ki.

Az elektronszerkezet megadásában segítségünkre van néhány **szabály**, alapelv.

Az **energiaminimum elve** szerint az elektronok mindig úgy épülnek be az elektronszerkezetbe, hogy a legkisebb energiájú szabad helyet foglalják el. A feltöltődés sorrendjét az **(n+l) szabály** határozza meg, miszerint az energiaszintek növekvő sorrendje azonos az (n+l) összeg növekvő sorrendjével. Azonos (n+l) értékek esetén a kisebb n értékű állapot az alacsonyabb energiájú.

A **Pauli-féle tilalmi elv** szerint valamely atomban két vagy több elektronnak sohasem lehet mind a négy kvantumszáma (n, l, m<sub>l</sub>, m<sub>s</sub>) azonos. Vagyis az azonos mellékkvantumszámú pályákon lévő elektronok száma  $4 \cdot l + 2$ . Az n. héjon az elektronok maximális száma  $2n^2$  (n a főkvantumszám).

Az energiaminimum és a Pauli-féle tilalmi elv figyelembevételével megmondható, hogy hová kerülnek az egyes elektronok az atomon belül és hogyan építik fel az elektronszerkezetet.

A **Hund-féle maximális multiplicitás elv** alapján az atom energiája akkor minimális, ha a párosítatlan elektronok (melyek azonos spinűek) száma maximális. [49]

Az atom helyét a **periódusos rendszerben** a rendszáma (protonok száma) határozza meg. A periódusos rendszerben a sorokat **periódusoknak**, az oszlopokat **csoporthoz** nevezzük. Az „a” betűvel jelölt csoportok a **főcsoportok**, míg a „b” jelű csoportok a **mellékcsoportok**. A periódusos rendszer **8 főcsoportból**, **8 mellékcsoportból** és **7 periódusból** áll.

A külső héj elektronjainak a száma a főcsoportszámot, míg az elektronszerkezet száma a periódusszámot adja meg. Az atomok kémiai reakciókban való viselkedését a vegyértékelektronok határozzák meg. Ezzel magyarázható, hogy az egy oszlopban található elemek hasonló kémiai tulajdonságúak.

Az **oxidációs szám** értéke valamely atom névleges, vagy valódi töltésének számértékével egyenlő. Az oxidációs szám mindig egyetlen atomra vonatkozik. Az elemek oxidációs száma mindig nulla. A molekulában kötött atom oxidációs száma a névleges töltésének a számértékével egyenlő. Egy molekulán belül az egyes atomok oxidációs számainak indexszámokkal szorzott összege mindig zérus.

A kötésben levő atom elektronvonzó képességét jellemzi az **elektronegativitás**. A programban az elektronegativitás **Pauling** számítása szerinti értékei kerültek felhasználásra.

### IV.3. Feladatok kvantumszámokra

Ebben a fejezetben az elkészült program az előző részben felvázoltak alapján úgy ábrázolja az ismereteket tények és szabályok formájában, hogy a következő kérdésekre válaszokat tud szolgáltatni:

- Milyen kvantumszámú héjak léteznek?
- Egy adott héjnak milyen alhéjai lehetnek?
- Az adott héjnak hány darab elektronpályája van?
- Az adott alhéjnak hány darab elektronpályája van?
- Az adott héjnak hány darab elektronja lehet maximálisan?
- Az adott alhéjnak hány darab elektronja lehet maximálisan?
- Az adott héj, alhéj esetén a Pauli-féle tilalmi elv szerint milyen mágneses kvantumszámok és mágneses spinquantumszámok lehetségesek?

A különböző kvantumszámok lehetséges értékeit az alábbi tényekkel, szabályokkal lehet megadni.

```
fő(1). fő(2). fő(3). fő(4). fő(5). fő(6). fő(7).
mellék(L):- fő(N), L is N-1.
mágneses(M):- mellék(M), M>0.
mágneses(M):- mellék(N), M is -N.
spin(0.5). spin(-0.5).
```

A héjak, alhéjak, valamint a héjakon, alhéjakon lehetséges pályák, elektronok számának a kvantumszámokkal való összefüggéseit az alábbi szabályokkal lehet megadni.

```

héj(N) :- fő(N) .
alhéj(N,L) :- fő(N), mellék(L), L<N.
héj_pályák(N,Db) :- fő(N), Db is N*N.
héj_elektronjai(N,Edb) :- fő(N), Edb is 2*N*N.
alhéj_pályák(L,Db) :- mellék(L), Db is 2*L+1.
alhéj_elektronjai(L,Edb) :- mellék(L), Edb is 4*L+2.
kvantumok(N,L,M,S) :- fő(N), mellék(L), L<N, mágneses(M), M>= (-L),
M<L+1, spin(S) .

```

### Néhány lehetséges célállítítás:

Amennyiben ezt a programot konzol alkalmazásként futtatjuk, akkor a következő célállításokat is megfogalmazhatjuk. Amennyiben meg szeretnénk adni egy célállításhoz az összes megoldást egyszerre, érdemes a kiírás után a fail paranccsal állandó visszalépésre készíteni a programot. A **write** (kiírás) és **nl** (új sor) parancsok segítségével értelmesen tördelve tudathatjuk az egyes megoldásokat a felhasználókkal.

**héj(J), write(J), nl, fail.**

**alhéj(3,L), write(3), write(' '), write(L), nl, fail.**

**héj\_pályák(3,Db).**

**héj\_pályák(N,Db), write(N), write(' '),write(Db), nl, fail.**

**héj\_elektronjai(4,Edb) .**

**héj\_elektronjai(N,Db), write(N), write(' '),write(Db), nl, fail.**

**alhéj\_pályák(3,Db).**

**alhéj\_pályák(N,Db), write(N), write(' '),write(Db), nl, fail.**

**alhéj\_elektronjai(4,Edb) .**

**alhéj\_elektronjai(N,Db), write(N), write(' '),write(Db), nl, fail.**

**kvantumok(2,1,M,S), write(M), write(' '), write(S), nl, fail.**

**kvantumok(2,L,M,S), write(L), write(' '),write(M), write(' '), write(S), nl, fail.**

**kvantumok(N,L,M,S), write(N), write(' '), write(L), write(' '), write(M),**

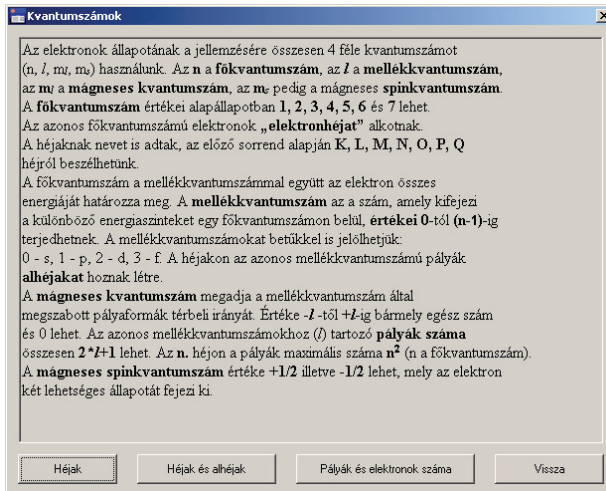
**write(' '), write(S), nl, fail.**



Ma már vannak olyan Prolog megvalósítások is, amelyek objektumorientált programok előállítására is képesek. Ezek a programváltozatok ötvözik a logikai programozás előnyeit az objektumorientált programozással. A főablakon a **Kvantumszámok** nyomógomb segítségével a 11. ábrán látható panelt kapjuk.

11. ábra

***Kvantumszámok, héjak és alhéjak jellemzői,  
Héjak és alhéjak pályáinak, elektronjainak a száma***



Forrás: saját

A **Héjak** nyomógomb segítségével juthatunk a 12. ábrán bemutatott ablakhoz. Ez az ablak a héjakról szolgáltat fontos információkat. A **Héjak és alhéjak** nyomógomb az 13. ábrán bemutatott panelhez vezet.

Ezen a panelen egy héj (5-ös főkvantumszámú) kiválasztása után megkaphatjuk, hogy milyen alhéjak lehetnek az adott héjon. Ezt a magyarázó szöveg is alaposan részletezi. A **Pályák és elektronok száma** nyomógomb segítségével pedig a 14. ábrán látható párbeszédablakhoz juthatunk.

12. ábra

## Héjak ablak

Az atommag körül lévő elektronok összességét **elektronburoknak** nevezzük. Az elektronok rétegesen, **héjakat** alkotva építik fel az elektronburokot. A **főkvantumszám** értékei alapállapotban 1, 2, 3, 4, 5, 6 és 7 lehet. Az azonos **főkvantumszámú** elektronok „**elektronhéjat**” alkotnak. A héjaknak nevet is adtak, az előző sorrend alapján **K, L, M, N, O, P, Q** héjról beszélhetünk.

Forrás: saját

13. ábra

## Héjak és alhéjak ablak

Az elektronok rétegesen, héjakat alkotva építik fel az **elektronburokot**. A **főkvantumszám** értékei alapállapotban 1, 2, 3, 4, 5, 6 és 7 lehet. Az azonos főkvantumszámú elektronok „**elektronhéjat**” alkotnak. A mellékquantumszámokat betűkkel is jelölhetjük: 0 - s, 1 - p, 2 - d, 3 - f. A héjakon az azonos mellékquantumszámú pályák **alhéjakat** hoznak létre. Azok a csoportok tartoznak egy **mezőbe** (s-mező, p-mező, d-mező, f-mező), amelyekben ugyanaz az **alhéj** kiépülése történik.

Héj (n):  Alhéj (l):  Kérem a héj kiválasztását!

Forrás: saját

A 14.ábrán egy adott héjhoz (6-os főkvantumszámú) tartozó lehetséges alhéjak (mellék-kvantumszámok), az alhéjakhoz tartozó elektronpályák száma, valamint a pályákon a maximális elektronszám értékei jeleníthetők meg.

14. ábra

*Pályák és elektronok száma ablak*

Héj (n)	Alhéj (l)	pályák	elektronok
6	0	1	2
6	1	3	6
6	2	5	10
6	3	7	14
6	4	9	18
6	5	11	22

Kérem a héj kiválasztását!

megjelenít

vissza

A **Pauli-féle tilalmi elv** szerint valamely atomban két vagy több elektronnak sohasem lehet mind a négy kvantumszáma ( $n, l, m_l, m_s$ ) azonos. Mivel az azonos **mellékvantumszámú** pályák száma  $2 \cdot l + 1$  ezért az azonos **mellékvantumszámú** pályákon lévő elektronok száma  $4 \cdot l + 2$  ( $l$  a mellékvantumszám). Mivel az  $n$ . héjon a pályák száma  $n^2$  ezért az  $n$ . héjon az elektronok maximális száma  $2n^2$  ( $n$  a főkvantumszám).

Forrás: saját

#### IV.4. Feladatok elemek elektronszerkezetére

Ebben a fejezetben az elkészült program a periódusos rendszer tetszőlegesen kiválasztott elemére megadja az elektronkonfigurációt, vagyis azt, hogy az adott elem alapállapotú atomjának elektronjai milyen elektronpályákat töltenek be, illetve, hogy melyik elektronpályán mennyi elektron található.

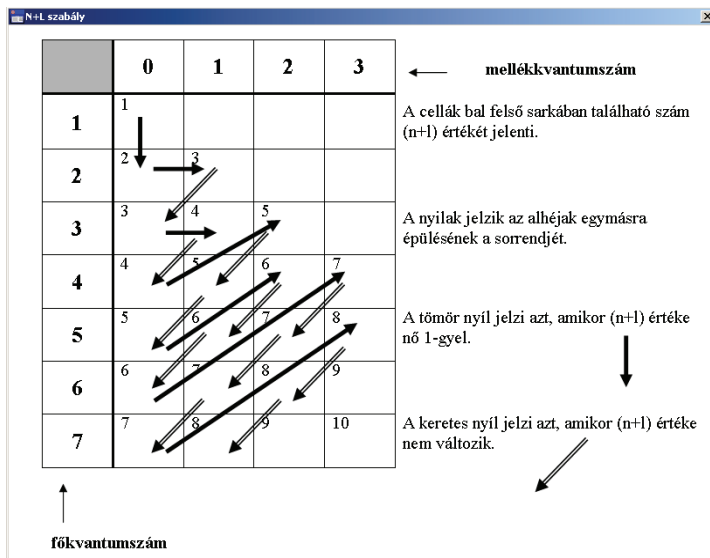
### Tények és szabályok:

```
fő(1). fő(2). fő(3). fő(4). fő(5). fő(6). fő(7).  
mellék(L):- fő(N), L is N-1.  
mágneses(M):- mellék(M), M>0.  
mágneses(M):- mellék(N), M is -N.  
spin(0.5). spin(-0.5).  
  
elektronpálya(E,N,L):- E>0, N<8, darab(L,E,Eb,E3),  
                                write(N), write(' '),  
                                write(L), write(' '),  
                                write(Eb), nl,  
                                változtat(N,L,N2,L2), !,  
                                elektronpálya(E3,N2,L2).  
  
elektronpálya(_,N,_):- N=8.  
elektronpálya(E,_,_):- E=0.  
  
darab(L,E,Eb,E3):- Db is 4*L+2, E>=Db, Eb is Db, E3 is E-Db.  
darab(L,E,Eb,E3):- Db is 4*L+2, E<Db, Eb is E, E3 is 0.  
  
változtat(N,L,N2,L2):- L=0, Osszeg is N+1,  
                                L2 is N // 2, N2 is Osszeg-L2.  
változtat(N,L,N2,L2):- L>0, N2 is N+1, L2 is L-1.
```

**Egy lehetséges célállítás:** *elektronpálya(67,1,0).*

Az **elektronpálya** szabályt, ha ezekkel a paraméterekkel meghívjuk, akkor a program kiírja, hogy 67 elektron hogyan helyezhető el az elektronpályákon a természeti törvényeknek megfelelően. Ez a programrészlet felhasználja az energiaminimum elvét, valamint az elektronburok kiépülésének  $(N+L)$  szabályát.

A 15. ábra az  $(N+L)$  szabály figyelembevételével az alhéjak felépítésének a sorrendjét szemlélteti.

*N+L szabály*

Forrás: saját

A 16. ábrán egy lehetséges objektumorientált elrendezést láthatunk, melyen megadott rendszámú (80) elem alapállapotú elektronkonfigurációja látható.

## Adott elem alapállapotú elektron-konfigurációja

**Elemek elektronszerkezete**

Az elektronszerkezet megadásában segítségünkre van néhány szabály, alapelv.  
 Az **energiaminimum elve** szerint az elektronok mindig úgy épülnek be az elektronhéjba, hogy a **legkisebb energiájú szabad helyet foglalják el**.  
 A feltöltődés sorrendjét az **(n+l) szabály** határozza meg, miszerint az energiaszintek növekvő sorrendje azonos az (n+l) összeg növekvő sorrendjével. Azonos (n+l) értékek esetén a kisebb n értékű állapot az alacsonyabb energiájú.  
 A **Pauli-féle tilalmi elv** szerint valamely atomban két vagy több elektronnak sohasem lehet mind a négy kvantumszáma (n, l, m<sub>l</sub>, m<sub>s</sub>) azonos.  
 Vagyis az azonos mellékkvantumszámú pályákon lévő elektronok száma  $4 \cdot l + 2$ . Az n. héjon az elektronok maximális száma  $2n^2$  (n a főkvantumszám).  
 Az energiaminimum és a Pauli-féle tilalmi elv figyelembevételével megmondható, hogy hová kerülnek az egyes elektronok az atomon belül és hogyan építik fel az elektronszerkezetet. A Hund-féle maximális multiplicitás elve alapján az atom energiája akkor minimális, ha a párosítatlan elektronok (melyek azonos spinűek) száma maximális.

Héj    Alhéj    Elektronok száma    Kérem az elem rendszámát:

1	0	2
2	0	2
2	1	6
3	0	2
3	1	6
4	0	2
3	2	10
4	1	6
5	0	2
4	2	10
5	1	6
6	0	2
4	3	14
5	2	10

Számol

Kivételes elemek

Vissza

Forrás: saját

Meg kell jegyezni azonban, hogy vannak kivételek is a szabály alól. A ma ismert **111 elem közül 20 elem** elektronszerkezete **pontosan nem adható meg ezekkel a szabályokkal**. Az eltérés nem nagy, csupán 1-2 elektron helyét találták kísérletileg más helyen. Ezen elemek mindegyike a **d** illetve az **f-mezőben** található. A nem-szabályos viselkedést több, eddig figyelembe nem vett módosító tényező befolyásolja. Bizonyos elemeknél (Cr, Cu, Mo, Ag, Au) kísérletileg nem az elektronszerkezet szabályos feltöltődését találták, hanem a félig ( $d^5$ ) illetve a teljesen ( $d^{10}$ ) feltöltött d-alhéjakat az előzőleg már betöltött s alhéj rovására.

Anomáliát észleltek az f-alhéjak kiépülésében is: a  $6s^2$  és a  $7s^2$  szintek után nem az (n-2)f nivó kiépülése kezdődik meg, hanem előbb az (n-1)d nivóé egy-egy elektronnal (La, Ac), ezután folytatódik az (n-2)f szintek betöltése 14-14 elektronnal, majd a már megkezdett (n-1)d alhéjak következnek.[50]

A 17. ábrán a kivételes elemek láthatóak a periódusos rendszerben szürke, sátirozott jelöléssel.

17. ábra

### Kivételes elemek

Kivételes elemek																	
1 a	2 a	3 a	4 a	5 a	6 a	7 a	8 a										
1 H hidrogén	2 He hélium																
3 Li lítium	4 Be beryllium											5 B bor	6 C szén	7 N nitrogén	8 O oxigén	9 F fluor	10 Ne neon
11 Na nátrium	12 Mg magnézium	13 Al alumínium	14 Si szilícium	15 P foszfor	16 S kén	17 Cl klóroz	18 Ar argon										
19 K kálium	20 Ca kalcium	21 Sc skandium	22 Ti titanium	23 V vanádium	24 Cr króm	25 Mn mangán	26 Fe vas	27 Co kobalt	28 Ni nikkel	29 Cu réz	30 Zn cink	31 Ga gallium	32 Ge germánium	33 As arszén	34 Se szelén	35 Br brom	36 Kr kréon
37 Rb rubídium	38 Sr sztrontium	39 Y yttrium	40 Zr cirkónium	41 Nb nióbium	42 Mo molibdén	43 Tc technécium	44 Ru rózium	45 Rh rózium	46 Pd palládium	47 Ag ezüst	48 Cd kadmium	49 In indium	50 Sn olaj	51 Sb antimon	52 Te tellúr	53 I jód	54 Xe xenon
55 Cs caesium	56 Ba barium	57-71 földalatti	72 Hf hafnium	73 Ta tantalum	74 W volfrám	75 Re renádium	76 Os osmium	77 Ir irídium	78 Pt platina	79 Au arany	80 Hg ezüstmeg	81 Tl tallium	82 Pb olaj	83 Bi bismut	84 Po polónium	85 At aszén	86 Rn radon
87 Fr francium	88 Ra rádium	89-103 földalatti	104 Rf rutherfordium	105 Db dubnium	106 Sg seaborgium	107 Bh bohrium	108 Hs hassium	109 Mt moscovium	110 Ds darmstadtium	111 Rg roentgenium							
71 La lantan	72 Ce ce	73 Pr praseodim	74 Nd neodim	75 Pm prometium	76 Sm samár	77 Eu europium	78 Gd gadolinium	79 Tb terbium	80 Dy dysprosium	81 Ho holmium	82 Er er	83 Tm terbium	84 Yb ytterbium	85 Lu lutetium			
89 Ac aktin	90 Th török	91 Pa protaktin	92 U urán	93 Np neptunium	94 Pu plutónium	95 Am amerícium	96 Cm kurium	97 Bk berkelium	98 Cf california	99 Es einsteinium	100 Fm fermium	101 Md mendelevium	102 No nobelium	103 Lr lawrencium			

Meg kell jegyezni azonban, hogy vannak kivételes is a szabály alól.  
A ma ismert **111 elem közül 20 elem** elektronszerkezete **pontosan nem adható ezekkel a szabályokkal**. Ezen elemek mindegyike a d illetve az f-mezőben található.  
A nem-szabályos viselkedést több, eddig figyelembe nem vett módosító tényező befolyásolja. Bizonyos elemeknél (Cr, Cu, Mo, Ag, Au) kísérletileg nem az elektrópályák szabályos feltöltődését találták, hanem a félg (d<sup>5</sup>) illetve a teljesen (d<sup>10</sup>) feltöltött d-alhéjakat az előzőleg már betöltött s alhéj rovására.  
Anomáliát észleltek az f-alhéjak kiépülésében is: a  $6s^2$  és a  $7s^2$  szintek után nem az (n-2)f nivó kiépülése kezdődik meg, hanem előbb az (n-1)d nivóé egy-egy elektronnal (La, Ac), ezután folytatódik az (n-2)f szintek betöltése 14-14 elektronnal, majd a már megkezdett (n-1)d alhéjak következnek.

Forrás: saját

Ennek az intelligens tananyagnak nemcsak az a nagy erénye, hogy a szabályokat követő elemek elektronszerkezetét pontosan meg tudja határozni, hanem az is, hogy rá tudja irányítani a figyelmet a „nem-szabályosan” viselkedő kivételes elemekre is. Ezek ismeretében pedig motiváló szerepe is van, ugyanis a kivételek szabályszerűségeinek a kutatására ösztönöz.

#### **IV.5. Feladatok az oxidációs számok és az elektronegativitás meghatározására**

Ebben a fejezetben az elkészült program egy tetszőlegesen kiválasztott elemre megadja az elektronegativitást, valamint a jellemző oxidációs számokat. Emellett a következő kérdésekre is válaszokat tud szolgáltatni:

- Mennyi egy megadott elem elektronegativitása?
- Megadott elemek közül melyik elektronegativitása a nagyobb?
- Egy megadott csoport (fő- és mellékcsoport) elemeinek milyen elektronegativitás értékei vannak?
- Hogyan változik az elektronegativitás értéke perióduson, csoporton belül?
- Amennyiben két adott elem között kémiai kötés jön létre, az milyen típusú lesz (ionos, kovalens, fémes)?
- Egy megadott elemnek milyen jellemző oxidációs számai lehetnek?
- Egy megadott csoport (fő- és mellékcsoport) elemeinek milyen jellemző oxidációs száma(i) lehet(nek)?

Az elemek adatai tényekként kerültek rögzítésre. Az **elem** egy hatargumentumos tényállítást, mely tartalmazza a következő ismereteket a következő sorrendben:

- az elem rendszáma (a periódusos rendszerben),
- az elem neve,
- a periódusának a sorszáma,
- az elem csoportjának a száma,
- a csoport főcsoport (a jelű), vagy mellékcsoport (b jelű),
- az elem elektronegativitása (Pauling szerint).



```

elem(1,`hidrogén`,1,1,`a`,2.1). elem(2,`hélium`,1,8,`a`,0).
elem(3,`lítium`,2,1,`a`,1.0). elem(4,`berillium`,2,2,`a`,1.5).
elem(5,`bór`,2,3,`a`,2.0). elem(6,`szén`,2,4,`a`,2.5).
elem(7,`nitrogén`,2,5,`a`,3.0). elem(8,`oxigén`,2,6,`a`,3.5).
elem(9,`fluor`,2,7,`a`,4.0). elem(10,`neon`,2,8,`a`,0).
elem(11,`nátrium`,3,1,`a`,0.9). elem(12,`magnézium`,3,2,`a`,1.2).
elem(13,`alumínium`,3,3,`a`,1.5).
elem(14,`szilícium`,3,4,`a`,1.8). elem(15,`foszfor`,3,5,`a`,2.1).
elem(16,`kén`,3,6,`a`,2.5).
elem(17,`klór`,3,7,`a`,3.0). elem(18,`argon`,3,8,`a`,0).
elem(19,`kálium`,4,1,`a`,0.8). elem(20,`kalcium`,4,2,`a`,1.0).
elem(21,`szkandium`,4,3,`b`,1.3). elem(22,`titán`,4,4,`b`,1.6).
elem(23,`vanádium`,4,5,`b`,1.6). elem(24,`króm`,4,6,`b`,1.6).
elem(25,`mangán`,4,7,`b`,1.5). elem(26,`vas`,4,8,`b`,1.8).
elem(27,`kobalt`,4,8,`b`,1.8). elem(28,`nikkel`,4,8,`b`,1.8).
elem(29,`réz`,4,1,`b`,1.9). elem(30,`cink`,4,2,`b`,1.6).
elem(31,`gallium`,4,3,`a`,1.6). elem(32,`germánium`,4,4,`a`,1.8).
elem(33,`arzen`,4,5,`a`,2.0). elem(34,`szelén`,4,6,`a`,2.4).
elem(35,`bróm`,4,7,`a`,2.8). elem(36,`kripton`,4,8,`a`,0).
elem(37,`rubídium`,5,1,`a`,0.7). elem(38,`stroncium`,5,2,`a`,0.9).
elem(39,`ittrium`,5,3,`b`,1.3). elem(40,`cirkónium`,5,4,`b`,1.4).
elem(41,`nióbium`,5,5,`b`,1.6). elem(42,`molibdén`,5,6,`b`,1.8).
elem(43,`technécium`,5,7,`b`,1.9).
elem(44,`ruténium`,5,8,`b`,2.2).
elem(45,`ródiium`,5,8,`b`,2.2). elem(46,`palládium`,5,8,`b`,2.2).
elem(47,`ezüst`,5,1,`b`,1.9). elem(48,`kadmium`,5,2,`b`,1.7).
elem(49,`indium`,5,3,`a`,1.7). elem(50,`ón`,5,4,`a`,1.8).
elem(51,`antimon`,5,5,`a`,1.9). elem(52,`tellúr`,5,6,`a`,2.1).
elem(53,`jód`,5,7,`a`,2.5). elem(54,`xenon`,5,8,`a`,0).
elem(55,`cézium`,6,1,`a`,0.7). elem(56,`bárium`,6,2,`a`,0.9).
elem(57,`lantán`,6,3,`b`,1.1). elem(72,`hafnium`,6,4,`b`,1.3).
elem(73,`tantál`,6,5,`b`,1.5). elem(74,`volfram`,6,6,`b`,1.7).
elem(75,`rénium`,6,7,`b`,1.9). elem(76,`ozmium`,6,8,`b`,2.2).
elem(77,`irídium`,6,8,`b`,2.2). elem(78,`platina`,6,8,`b`,2.2).
elem(79,`arany`,6,1,`b`,2.4). elem(80,`higany`,6,2,`b`,1.9).
elem(81,`tallium`,6,3,`a`,1.8). elem(82,`ólom`,6,4,`a`,1.8).
elem(83,`bizmut`,6,5,`a`,1.9). elem(84,`polónium`,6,6,`a`,2.0).
elem(86,`radon`,6,8,`a`,0). elem(87,`francium`,7,1,`a`,0.7).
elem(88,`rádiium`,7,2,`a`,1.1). elem(89,`aktínium`,7,3,`b`,1.1).

```

Az elemek oxidációs számának a megadásakor már nemcsak tényállításokat, hanem szabályokat is meg kell fogalmazni, mivel egy elemnek többféle oxidációs száma is lehet, továbbá vannak bizonyos szabályszerű összefüggések [51].

Az **oxidációs\_szá**m egy kétargumentumos szabály, mely tartalmazza a következő ismereteket:

- az elem neve,
- oxidációs számának értéke.

A következő szabályok szerint lehet megadni egy bizonyos csoport (a, b) jellemző oxidációs számait. Az első szabály azt rögzíti, hogy minden elemnek létezik a 0 oxidációs száma (ez akkor teljesül, ha az adott elem nem vegyületében, hanem elemi állapotban van). Ezek az oxidációs szám értékek a [52] forrás adatai szerint kerültek rögzítésre.

```
oxidációs_szá
```

Vannak azonban olyan oxidációs szám értékek is, amelyek egyedileg érvényesek az elemekre, ezeket célszerű tényekként (vagyis változót nem tartalmazó szabályokként) rögzíteni.

```

oxidációs_száma('tallium',1). oxidációs_száma('szén',-4).
oxidációs_száma('szén',2).
oxidációs_száma('ón',2). oxidációs_száma('ólom',2).
oxidációs_száma('nitrogén',4). oxidációs_száma('nitrogén',2).
oxidációs_száma('foszfor',4). oxidációs_száma('oxigén',-1).
oxidációs_száma('kén',2). oxidációs_száma('kén',6).
oxidációs_száma('szelén',6). oxidációs_száma('tellúr',6).
oxidációs_száma('polónium',2).
oxidációs_száma('klór',3). oxidációs_száma('klór',7).
oxidációs_száma('jód',7).
oxidációs_száma('réz',2). oxidációs_száma('cink',1).
oxidációs_száma('arany',3). oxidációs_száma('higany',1).
oxidációs_száma('vanádium',4). oxidációs_száma('niobium',3).
oxidációs_száma('mangán',4). oxidációs_száma('mangán',6).
oxidációs_száma('molibdén',2). oxidációs_száma('molibdén',3).
oxidációs_száma('molibdén',4). oxidációs_száma('molibdén',5).
oxidációs_száma('volfram',2). oxidációs_száma('volfram',3).
oxidációs_száma('volfram',4). oxidációs_száma('volfram',5).
oxidációs_száma('ruténium',3). oxidációs_száma('ruténium',6).
oxidációs_száma('ruténium',8).
oxidációs_száma('ródium',3). oxidációs_száma('rénium',6).
oxidációs_száma('rénium',4). oxidációs_száma('rénium',2).
oxidációs_száma('rénium',-1). oxidációs_száma('ozmium',3).
oxidációs_száma('ozmium',6). oxidációs_száma('ozmium',8).
oxidációs_száma('irídium',3). oxidációs_száma('irídium',6).

```

Miután sikerült rögzíteni az elemek adatait, nézzük, hogy milyen lényeges információkat tudunk kinyerni a tárolt adatokból. A további összefüggéseket érdemes szabályok formájában megfogalmazni, majd a Prolog számára kérdéseket, célállításokat megadni a konkrét esetekre vonatkozóan.

**A csoport\_számai** (háromargumentumos) szabály egy magadott csoport (pl. a 3.b) jellemző oxidációs számait szolgáltatja.

```

csoport_számai(A,B,C):- elem(_,X,_,A,B,_) , oxidációs_száma(X,C) .

```

A **hasonlit\_en** (kétargumentumos) szabály két megadott elem (pl. szén, hidrogén) elektronegativitás értékeit hasonlítja össze, majd kiírja a nagyobb elektronegativitás értékű elem nevét.

```
hasonlit_en(A,B):- elem(_ ,A,_ ,_ ,X), elem(_ ,B,_ ,_ ,Y), X>Y,
                    write(A), nl.
hasonlit_en(A,B):- elem(_ ,A,_ ,_ ,X), elem(_ ,B,_ ,_ ,Y), X=Y,
                    write(`Az elektronegativitásuk egyforma.`), nl.
hasonlit_en(A,B):- elem(_ ,A,_ ,_ ,X), elem(_ ,B,_ ,_ ,Y), X<Y,
                    write(B), nl.
```

Az **elektronegativitás** (kétargumentumos) szabály egy megadott elem (pl. a 80-as rendszámú higany) elektronegativitását szolgáltatja az **elem** szabály felhasználásával.

```
elektronegativitas(X,En):- elem(_ ,X,_ ,_ ,En).
```

Az **en\_csoport** (háromargumentumos) szabály egy megadott elemcsoport (pl. 1.a csoport) elektronegativitás értékeit, míg az **en\_periódus** (kétargumentumos) szabály egy megadott periódus (pl. a 4.) elektronegativitás értékeit szolgáltatja.

```
en_csoport(P,Q,En):- elem(_ ,_ ,P,Q,En).
```

```
en_periódus(S,En):- elem(_ ,S,_ ,_ ,En).
```

**Néhány lehetséges célállítás:**

```
csoport_számai(4, 'a', C), write(C), nl, fail.
csoport_számai(1, 'a', C), write(C), nl, fail.
hasonlit_en('szén', 'hidrogén').
hasonlit_en('hidrogén', 'hidrogén').
hasonlit_en('bróm', 'klór').
oxidációs_száma('szén', C), write(C), nl, fail.
elektronegativitas('szén', En).
en_csoport(5, 'a', En), write(En), nl, fail.
en_csoport(3, 'b', En), write(En), nl, fail.
en_periódus(5, En), write(En), nl, fail.
```

A 18. ábrán egy kiválasztott elem (17 – klór), valamint egy csoport (3.b) jellemző oxidációs számai láthatóak.

18. ábra

*Adott elem illetve csoport jellemző oxidációs számai*

**Oxidációs számok**

Megadott elem oxidációs számai

Kérem az elem rendszámát:

Jellemző oxidációs számai:

Az elem neve:

Megadott csoport oxidációs számai

Kérem a csoport kiválasztását:

Jellemző oxidációs számai:

Az **oxidációs szám** értéke valamely atom névleges, vagy valódi töltésének számértékével egyenlő. Az oxidációs szám mindig egyetlen atomra vonatkozik.  
 Az elemek oxidációs száma mindig nulla.  
 A molekulában kötött atom oxidációs száma a névleges töltésének a számértékével egyenlő.  
 Egy molekulán belül az egyes atomok oxidációs számainak indexszámokkal szorzott összege mindig zérus.

Forrás: saját

A 19. ábrán egy kiválasztott elem (80 – higany), valamint egy periódus (4) és egy csoport (1.a) elektronegativitás értékei láthatóak.

19. ábra

*Adott elem illetve csoport elektronegativitás értékei*

**Elektronegativitás**

Megadott elem elektronegativitása

Kérem az elem rendszámát:

Az elem neve:

Elektronegativitás értéke:

Megadott periódus, csoport elektronegativitás (EN) értékei

Kérem a periódus kiválasztását:

Kérem a csoport kiválasztását:

Elektronegativitás értékei:

A kötésben levő atom elektronvonzó képességét jellemzi az **elektronegativitás**.  
 A programban az elektronegativitás **Pauling** számítása szerinti értékei kerültek felhasználásra.

Forrás: saját

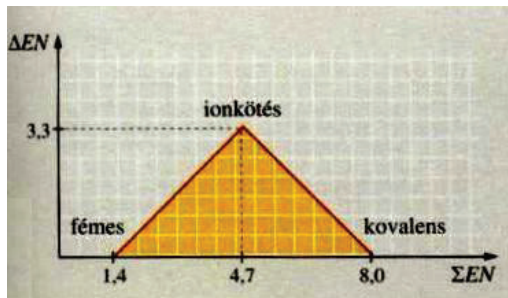
A kémiai kötések kialakításának a tulajdonságait sok tényező befolyásolja, melyek közül az egyik legfontosabb az elemek elektronvonzó képessége (elektronegativitása).

A **kötéstípus** (kétargumentumos) szabály különböző atomok esetén az elemek elektronegativitás értékei alapján megadja a létesülő kémiai kötés típusát. Ennek megfogalmazásakor felhasználásra kerültek a 20. ábrán bemutatott megfontolások. Ezek ugyanis figyelembe veszik az elemek elektronegativitásának az összegét, valamint a különbségét is. A programban a következőkben ismertetett határértékek tapasztalati úton lettek meghatározva. Amennyiben az elemek elektronegativitásának különbsége 2-nél nem kisebb, abban az esetben ionos jellegű kötés jön létre közöttük. Abban az esetben amennyiben ez a különbség 1,5-nél kisebb, akkor figyelembe kell venni az elektronegativitások összegét is. Ez, ha nem kisebb 5-nél, abban az esetben kovalens kötést, ha nem nagyobb 3-nál akkor pedig fémes kötést fog eredményezni. A köztes tartományban ( $3 < \Sigma EN < 5$ ) úgynevezett átmeneti jellegű kötés alakul ki.

Ezeket az ismereteket az alábbi szabályokkal lehet kifejezni:

```
kötéstípus(A,B):- elem(_,A,_,_,X), elem(_,B,_,_,Y), W is X-Y,
                    ((W>=2) ; (W<(-2))), write(`ionos kötés`),nl.
kötéstípus(A,B):- elem(_,A,_,_,X), elem(_,B,_,_,Y), Z is X+Y,
                    W is X-Y, Z>=5, W<1.5, W>(-1.5),
                    write(`kovalens kötés`),nl.
kötéstípus(A,B):- elem(_,A,_,_,X), elem(_,B,_,_,Y), Z is X+Y,
                    W is X-Y, 3>=Z, W<1.5, W>(-1.5),
                    write(`fémes kötés`),nl.
kötéstípus(_,_) :- write(`átmeneti kötés`),nl.
```

A negyedik **kötéstípus** szabálynál azért nem kell vizsgálni az elektronegativitás értékeket, mivel a Prolog feladatmegoldó algoritmus az azonos nevű szabályokat sorrendben próbálja felhasználni. Tehát ehhez a szabályhoz csak abban az esetben jut el, ha az előző 3 szabály már nem volt végrehajtható.

*Elektronegativitás és kötéstípus*

Forrás: [53]

Az oktatási segédanyag továbbfejlesztése természetesen tovább folyik, a fejlesztés irányait a kémiai kötések kialakításának tulajdonságai, a reakciókészség alapjai illetve a lehetséges reakciók megjósolásának, a létrejövő reakciók egyenleteinek meghatározásai jelentik.

A cél nemcsak egy olyan intelligens oktatási rendszer létrehozása, amely a jövő oktatási szoftverének igényeit is tartalmazza, hanem amely kémiai szakértői rendszerként, valamint elektronikus kézikönyvként is használható.

A tananyag alapjául szolgáló Prolog programot esettanulmányként a Mesterséges intelligencia, majd később a Logikai programozás kurzusaimon is felhasználtam. A program tervezésének, megvalósításának magyarázatokor jutottam arra a következtetésre, hogy egy ilyen bonyolult programot leginkább a programozott oktatás módszertanával lehetne elmagyarázni, apró lépésekre bontva ezt az igen összetett munkafolyamatot. Különösen akkor indokolt ennek a módszertannak a használata, ha e-learning rendszerben szeretnénk elhelyezni.

Így a következő lépés egy olyan programozott tananyag elkészítése volt, amely ennek a segédanyagnak az elkészítését tartalmazza, de egyszersmind a Prolog nyelv tanítását is nagyon jól tudja szolgálni. Ez a tananyag előre meghatározott lépésekben, rövid tanulási egységeként mutatja be egy Prolog alkalmazás elkészítését.

A tananyagot a tanulók egyéni módon, saját ütemezés szerint sajátíthatják el, ha szükségesnek érzik vissza is léphetnek és átismételhetik az előzetesen megismerteket. A tanulási egységek végén kérdések találhatók, melyek sikeres megválaszolása a továbblépéshez nélkülözhetetlen.

A programozott tananyag 34 lépésből áll, amelyek mindegyikét külön-külön egy új programablakban jeleníti meg. Amennyiben a tanuló úgy gondolja, hogy elsajátította az új ismereteket és a kérdésekre válaszolni tudott, akkor továbbléphet. Nagyon fontos a tanuló részéről az aktív közreműködés, vagyis valóban próbálja meg a kérdésekre a válaszokat megtalálni, csak azután lépjen tovább. Természetesen lehetőség van egy korábbi ablakra is visszalépni, valamint arra is, hogy a teljes mintaprogramot megjelenítse a tanuló a tanulási egységek bármelyikén.

A programozott tananyag ugyancsak Win-Prolog nyelven készült, de később a Moodle e-learning rendszerben lecke formátumban is elkészítésre került. Az e-learning rendszerben már az is lehetővé vált, hogy a tanulókat pontokkal mérjem, s a továbbhaladást ennek függvényében irányíthassam.

A teljes programozott tananyag a ***B függelékben*** található.



## V. Mesterséges intelligencia és Logikai programozás oktatása e-learning rendszerben

2008. tavaszától kezdve oktatok Mesterséges intelligencia szabadon választható C tárgyat a Károly Róbert Főiskolán. A nappali tagozaton heti 2 órában ismerkedhetnek meg hallgatóink a mesterséges intelligencia tárgykörével, legfontosabb területeivel. Tanulóink betekintést nyerhetnek a robotikába, megismerhetik a ma létező legkorszerűbb robotokat, fejlesztésük motivációit, problémáit. Ezen túl a szakértői rendszerek építésének, kialakításának munkafolyamatába is bepillanthatnak, valamint a logikai programozással, mint a legmagasabb szintű programozással, is megismerkedhetnek.

Az oktatási tevékenység hatékonyságának növelése érdekében Moodle e-learning rendszerrel támogatom ezt a kurzust. Mivel a heti rendszerességi kontakt órákon kívül e-learning-es támogatást is adok hallgatóimnak, ezért egy klasszikus *blended learning* környezetet sikerült így kialakítanom. A blended learning ugyanis egy olyan vegyes oktatási forma, amely a hagyományos oktatási formák mellett az online tanulási technikákat is alkalmazza. Ez a kombinált oktatási forma hatékonyan ötvözi mindkét változat előnyeit [42].

A Károly Róbert Főiskola Moodle e-learning rendszere a **<http://mentor.karolyrobert.hu>** [54] címen érhető el az iskola tanulói, oktatói számára. A kurzusok felvétele regisztrációköteles.

A Mesterséges intelligencia tárgyterülete nagyon széles, ezért az oktatási tapasztalataim alapján egy újabb tantárgyat is meghirdettem Logikai programozás címmel. Ezen a kurzuson belül még részletesebben tudom hallgatóimnak a Prolog nyelvet bemutatni, s a hallgatókkal közösen Prolog alkalmazásokat fejleszteni.

Az évek óta felhalmozott tapasztalatok alapján fogalmaztam meg a következő tézisemet, miszerint:

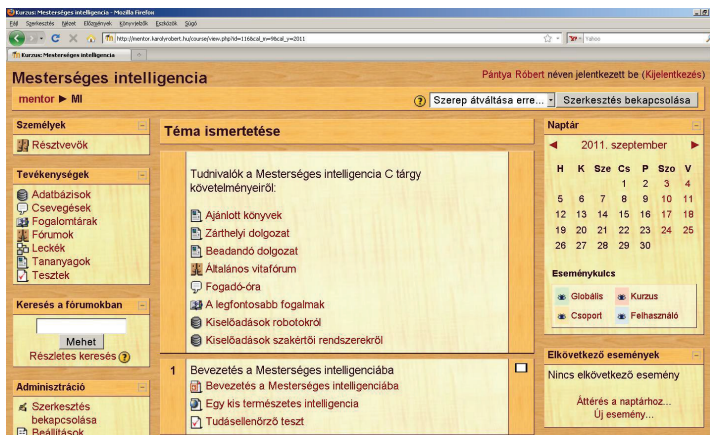
**A mesterséges intelligencia, valamint a logikai programozás oktatását e-learning rendszerrel célszerű támogatni, különösen nem-informatikus tanulók oktatása esetén.**

## V.1. A Mesterséges intelligencia oktatási lehetőségei e-learning rendszerben

A kurzus megjelenési formája látható a következő ábrán.

21. ábra

*A Mesterséges intelligencia kurzus*



Forrás: saját

A kurzus főablakán a bal felső sarokban a kurzus résztvevőit érhetjük el, üzeneteket, elektronikus levelet küldhetünk a hallgatóknak, vagy hallgatói csoportoknak. Megjelenik a főablakban egy naptár is, melyen a kurzuseseményeket jeleníthetjük meg (zárthelyi dolgozatok időpontjai, fogadórak rendje, vizsgaidőpontok, stb.). A központi, középső részben a kurzusszervezés egyéb fontos elemeit, valamint a tantárgy fő témaköreihez tartozó segédanyagokat adhatjuk meg.

A kurzus főablakán a „Tudnivalók a Mesterséges intelligencia C tárgy követelményeiről” részben kerültek elhelyezésre a legfontosabb technikai, kommunikációs és szervezési ismeretek. Általános vitaforum került üzemeltetésre, meghatározott időben fogadóórákat tartok, melyen a rendszer csevegő szolgáltatását, valamint a Skype szolgáltatást veszem igénybe.

A tantárgyhoz tartozó legfontosabb fogalmakat „*A legfontosabb fogalmak*” fogalomtárba gyűjtöttem össze. Ebből a fogalomtárból készíthetők olyan teszt-feladatsorok, melyek vizsga-tesztként használhatóak a számonkérés során. A hallgatóknak önállóan egy kiválasztott robotról, vagy megadott szakértői rendszerről kiselőadást kell készíteniük, melyet a „*Kiselőadások robotokról*”, illetve a „*Kiselőadások szakértői rendszerekről*” adatbázisokban kell elhelyezniük, s így mindenki számára elérhetővé tenniük.

A Mesterséges intelligencia tantárgy főbb témakörei:

- Bevezetés a Mesterséges intelligenciába
- Robotika
- Szakértői rendszerek
- Prolog alapok
- Feladatok megoldása Prolog-ban
- Rekurzivitás a Prolog-ban
- Listák és összetett kifejezések
- Objektum-orientált programozás WIN-Prolog-ban

A ***Bevezetés a Mesterséges intelligenciába*** fejezet megismerteti a hallgatót a mesterséges intelligencia tárgykörével, vizsgálódási módszereivel, történeti áttekintésével, alkalmazási lehetőségeivel.

A ***Robotika*** fejezetben a robotok típusaival, legnépszerűbb képviselőivel, felhasználási területeivel, fejlesztésük motivációival, problémáival ismerkedhetnek meg a hallgatók.

A ***Szakértői rendszerek*** fejezet a problémamegoldás legkorszerűbb technikáival, a tudás adott formába öntésével (tudásreprezentáció) és különböző szakértői rendszer típusokkal (orvosi, kémiai, üzleti, informatikai szakértői rendszerek) ismerteti meg. A hallgatók a szakértői rendszerek építésének, kialakításának munkafolyamatába is bepillanthatnak ebben a részben.

A ***Prolog alapok*** fejezetben kezdődik el a logikai programozás megismerése. Ehhez a Win-Prolog [46] programnyelvet használom. Ebben a fejezetben megismerik tanulóink a Prolog nyelvi, szintaktikai alapjait néhány egyszerű példa alapján.

A **Feladatok megoldása Prolog-ban** fejezet célja az előző fejezetben megtanultak gyakoroltatása mintafeladatok segítségével.

A **Rekurzivitás a Prolog-ban** fejezet már bonyolultabb programszerkezetek megismertetését is célul tűzi ki. A rekurzió megfelelő begyakoroltatását ebben a részben is mintafeladatok, valamint az online logikai játékok segítik.

A **Listák és összetett kifejezések** fejezet célja megismertetni a hallgatót a bonyolultabb adatszerkezetek kezelésével.

Végül az **Objektum-orientált programozás WIN-Prolog-ban** fejezet megismerteti a tanulót a vizuális programelemek használatával logikai programozási környezetben. Ez a rész azért is nagyon fontos, mert a korábbi Prolog verziók nem rendelkeztek objektum-orientált kiterjesztéssel, így a tanulókat nem volt könnyű motiválni. Ebben a Prolog verzióban azonban már nagyon könnyen lehet objektum-orientált programokat készíteni.

Minden témakörhöz tartoznak leckék (📖), letölthető fájlok, melyek lehetnek pdf dokumentumok (📄), saját készítésű videók (📺) a programnyelv használatához, a programok forrás fájllai (📄). A tananyag törzsanyaga leckékben, programozott oktatás segítségével kerültek kifejtésre. Minden témakör végén „Tudásellenőrző teszt”-ek (📝) kerültek elhelyezésre.

Nagyon fontos megjegyezni azt, hogy a megoldott teszt eredményeit csak a hallgató, valamint az oktató láthatja. A teszteket lehetősége van a hallgatónak többször is gyakorolni, akár más időpontokban is újra kitölteni (természetesen más kérdésekkel egy újabb tesztet). A tanár táblázatos formában láthatja a próbálkozások számát, a hallgatók pontjait, melynek segítségével számára is visszajelzés érkezik a tananyag megoldhatóságával, feldolgozhatóságával kapcsolatban. Ezeket az eredményeket pedig statisztikailag is tudja elemezni.

A következő ábrán a kurzus témaköreinek részletes listája látható, ahogyan azt a hallgatók láthatják, valamint ahogyan elérhetik azokat.

## 22. ábra

1	Bevezetés a Mesterséges intelligenciába Bevezetés a Mesterséges intelligenciába Egy kis természetes intelligencia Tudásellenőrző teszt	5	Feladatok megoldása Prolog-ban Gyakorlatok 1 Virágok ültetése Családfa Gyilkosság Ház körüli állatok Tudásellenőrző teszt
2	Robotika Robotika Robot foci 1 Robot foci 2 Robot foci 3 Robot - amerikai foci Albo Asimo Kismet 1 Kismet 2 Táncoló robotok Főző robot Háztartási robot Takarító robot 1 Takarító robot 2 Tudásellenőrző teszt	6	Rekurzívítás a Prolog-ban Rekurzív függvény-hívások Prolog-ban Gyakorlatok 2 Kibővített családfa Összegzés Hanoi tornyai (online játék) 1 Hanoi tornyai (online játék) 2 Hanoi tornyai (program) Tudásellenőrző teszt
3	Szakértői rendszerek Szakértői rendszerek Tudásellenőrző teszt	7	Listák és összetett kifejezések Listák és összetett kifejezések kezelése Prolog-ban Elektronszerkezet (feladat) Elektronszerkezet (program) Tudásellenőrző teszt
4	Prolog alapok Programozás Prolog-ban Kezdeti lépések Prolog-ban Fagyalt-vásárlás Tudásellenőrző teszt	8	Objektum-orientált programozás WIN-Prolog-ban Objektum-orientált programok készítése WIN-Prologban A Dialog Editor használata Objektum-orientált elemek Elektronszerkezet (objektum-orientált feladat) Elektronszerkezet (objektum-orientált program) Tudásellenőrző teszt

Forrás: saját

A tanulók motivációját nagyban lehet növelni olyan videókkal (pl.: robotokról), amelyeket különböző videó-megosztókon, pl.: YouTube, IndaVideo, stb., tettek közzé. A keretrendszerbe beágyazásra került számos ilyen oktató videó pl.: többféle robot-fociról, Aibo-ról, Asimo-ról, Kismet-ről, valamint különféle háztartási robotokról (takarító, főző robotok).

Screen recorder programokkal (pl. Camtasia) könnyen készíthetők olyan videók, amelyek az adott programozási nyelv alaptechnikáival ismerteti meg a tanulót. A keretrendszerbe számos ilyen saját készítésű videó került beillesztésre („Kezdeti lépések Prolog-ban”, „Dialog Editor használata”, „Objektum-orientált elemek”, stb). Ezek a fájlok tartalmazzák pl.: a program elindítását, fordítását, szintaktikai ellenőrzését, stb.

Az Interneten nagyon sok helyen lehet találni online logikai játékokat is. Érdemes elérhetővé tenni kipróbálásra, mivel nagyon fontos, hogy ezek az ismeretek programíráskor már ismertek legyenek. A Moodle-ben könnyen egybegyűjthetőek ezek a web-címek.

A nehezebb, problémásabb részeket leckékben, programozott oktatás segítségével fejtettem ki. Ilyen leckék pl.: „*Programozás Prologban*”, „*Rekurzív függvényhívások Prologban*”, „*Listák és összetett kifejezések kezelése Prologban*”, valamint az „*Objektum-orientált programok készítése WIN-Prologban*”. Ezekben a leckékben ellenőrző kérdések, valamint a választóktól függően elágazási pontok vannak elhelyezve. Amennyiben a tanuló nem tudja kiválasztani a helyes választ, abban az esetben egy részletesebb, kisebb lépésekben haladó útvonalra kerül. A helyes válasz megadása után haladhat tovább a lecke feldolgozásával.

A tudásellenőrzésnek és az értékelésnek többféle módzata van a Moodle rendszerben. Ezen módszerek fontos képviselői a tesztek. A tesztek nagyon hasznos eszközöknek bizonyulnak a hallgatók tudásának a mérésére. Az önálló tanulást nagyban elősegítik ezek a visszacsatolások. A teszteknek többféle fajtáját különbözteti meg ez az e-learning rendszer, így pl. használhatunk feleletválasztós, kiegészítő, számjegyes, igaz-hamis, párosító, számítási, esszé, stb. típusú teszteket.

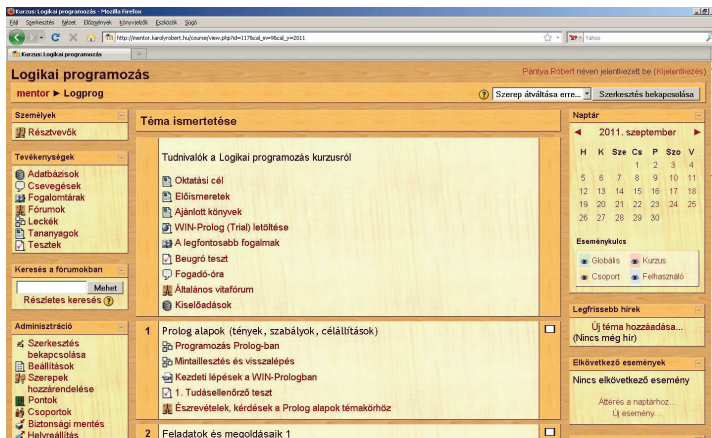
## **V.2. A Logikai programozás oktatási lehetőségei e-learning rendszerben**

A Logikai programozás kurzus célja a problémamegoldó képesség fejlesztése [55]. A tárgy oktatása során célként tűzöm ki, hogy megismertessem a hallgatót a deklaratív programozás rejtelseivel a Win-Prolog programnyelv segítségével [46]. A kurzus a Prolog nyelv elméleti tananyagaként Márkus Zs. [23] munkájára épít, valamint felhasználja Makány Gy. [22] és Máday Á. [56] munkáit a gyakorlati feladatok forrásaként. A kurzus során feladattípus-orientált programozás-tanítási módszerrel történik a programozás oktatása.

Emellett alkalmazásra kerül a mintapéllda alapján való módszer is az esettanulmányok bemutatása, illetve a nagyszámú feladat megoldása során [57]. Mivel a logikai programozás ismereteinek elsajátítása nem egyszerű feladat, ezért különösen fontos ezeknek az eszközöknek a bevezetése ebben a témában. A kurzus főablaka a 23. ábrán látható.

23. ábra

### *A Logikai programozás kurzus főablaka*



Forrás: saját

A 24. ábrán a kurzus témaköreinek részletes listája látható, ahogyan azt hallgatóink láthatják, valamint ahogyan elérhetik azokat.

<p>1 Prolog alapok (tények, szabályok, célállítások)</p> <ul style="list-style-type: none"> <li>Prologozás Prolog-ban</li> <li>Mintaillesztés és visszalépés</li> <li>Kezdeti lépések a WIN-Prologban</li> <li>1. Tudásellenőrző teszt</li> <li>Észrevételek, kérdések a Prolog alapok témakörhöz</li> </ul>	<p>4 Feladatok és megoldásaik 2</p> <ul style="list-style-type: none"> <li>Feladatok 2</li> <li>Nagycsalád (kibővített családja) mintafeladat megoldása lépésekben</li> <li>Nagycsalád (kibővített családja)</li> <li>Összegzés</li> <li>Hanoi tornyai (online játék) 1</li> <li>Hanoi tornyai (online játék) 2</li> <li>Hanoi tornyai (program)</li> <li>4. Tudásellenőrző teszt</li> <li>Észrevételek, kérdések a Feladatok és megoldásaik 2 témakörhöz</li> </ul>
<p>2 Feladatok és megoldásaik 1</p> <ul style="list-style-type: none"> <li>Feladatok 1</li> <li>Fagyfali-vásárlás mintafeladat megoldása lépésekben</li> <li>Virágok ültetése</li> <li>Családfa</li> <li>Gyilkosság</li> <li>Ötthoni állapotok</li> <li>Kedvenc játékok</li> <li>2. Tudásellenőrző teszt</li> <li>Észrevételek, kérdések a Feladatok és megoldásaik 1 témakörhöz</li> </ul>	<p>6 Listák és összetett kifejezések</p> <ul style="list-style-type: none"> <li>Listák és összetett kifejezések kezelése Prolog-ban</li> <li>Elemek elektronszerkezete esettanulmány</li> <li>Elektronszerkezet (feladat)</li> <li>Elektronszerkezet (program)</li> <li>5. Tudásellenőrző teszt</li> <li>Észrevételek, kérdések a Listák és összetett kifejezések témakörhöz</li> </ul>
<p>3 Rekurzivitás a Prolog-ban</p> <ul style="list-style-type: none"> <li>Rekurzív függvény-hívások Prolog-ban</li> <li>Vágás (cut) művelet</li> <li>Tagadás</li> <li>Ciklus-szervezés rekurzíval</li> <li>3. Tudásellenőrző teszt</li> <li>Észrevételek, kérdések a Rekurzivitás a Prolog-ban témakörhöz</li> </ul>	<p>6 Objektum-orientált programozás WIN-Prologban</p> <ul style="list-style-type: none"> <li>Objektum-orientált programok készítése WIN-Prologban</li> <li>A Dialog Editor használata</li> <li>Objektum-orientált elemek készítése</li> <li>Elemek elektronszerkezete (objektum-orientált) esettanulmány</li> <li>Elektronszerkezet feladat (objektum-orientált)</li> <li>Elektronszerkezet program (objektum-orientált)</li> <li>6. Tudásellenőrző teszt</li> <li>Észrevételek, kérdések az Objektum-orientált programozás WIN-Prologban témakörhöz</li> </ul>

Forrás: saját

A **„Prolog alapok (tények, szabályok, célállítások)”** fejezetben kezdem el a logikai programozás ismertetését. Ebben a fejezetben megismerik tanulóink a Prolog nyelvi, szintaktikai alapjait néhány egyszerű példa alapján. Két leckét is tartalmaz ez a bevezető rész, melyek a programozott oktatás módszertanát követik. Camtasia screen recorder programmal készült a *„Kezdeti lépések a WIN-Prolog-ban”* nevű videó, amely a programozási nyelv alaptéchnikáival ismerteti meg a tanulót. A fejezet végén (csakúgy, mint a többi fejezetnél) teszttel történik a hallgató tudásának mérése, valamint fórum is üzemeltetésre kerül az észrevételek, kérdések megvitatására.

A **„Feladatok és megoldásaik 1”** fejezet célja az előző fejezetben megtanultak gyakoroltatása mintafeladatok segítségével. Pdf állományban megtalálhatóak a feladatok, s letölthetőek a forrás-programok.

A **„Rekurzívítás a Prolog-ban”** fejezet már bonyolultabb programszerkezetek megismertetését is célul tűzi ki. Mivel ez egy igencsak „problémás” területe a logikai programozás oktatásának, ezért itt került elhelyezésre a legtöbb lecke (*„Rekurzív függvény-hívások Prologban”, „Vágás (cut) művelet”, „Tagadás”, „Ciklusszervezés rekurzíval”*), itt érvényesül a legerőteljesebben a tanári irányító szerep a tananyag egyéni feldolgozásában.



A következő, **„Feladatok és megoldásaik 2”** fejezet célja itt is az előző fejezetben megtanultak gyakoroltatása mintafeladatok segítségével. A rekurzió megfelelő begyakoroltatását ebben a részben is kidolgozott feladatok segítik. Az egyik „leg híresebb” példának, a Hanoi tornyai feladatnak a megoldásához azonban segítenek az interneten elérhető online logikai játékok [58], [59]. Mivel a játékszabályokat programíráskor már ismertnek tekintjük, ezért érdemes elérhetővé tenni kipróbálásra ilyen formában ezeket a feladatokat. Tapasztalat szerint sokkal egyszerűbb ezeknek a feladatoknak a megoldása, ha már előtte játszhatnak vele, kipróbálhatják azokat. Számos egyéb logikai feladat online játék formájában is megtalálható az interneten (Nim játék, 8 királynő elhelyezése, 8-as, 15-ös kirakó játék, stb). A Moodle-ben könnyen egybegyűjthetők ezek a web-címek.

A **„Listák és összetett kifejezések”** fejezet célja megismertetni a hallgatót a bonyolultabb adatszerkezetek kezelésével. Ebben a fejezetben erősen támaszkodom egy korábban kidolgozott esettanulmányra (elemek elektronszerkezete) [60], melynek segítségével jól demonstrálhatóak a listák, valamint az összetett kifejezések használata.

Végül az **„Objektum-orientált programozás WIN-Prolog-ban”** fejezet megismerteti a tanulót a vizuális programelemek használatával logikai programozási környezetben. Ebben a fejezetben két video fájl is szemlélteti az új ismereteket (**„Dialog Editor használata”**, **„Objektum-orientált elemek készítése”**). Ez a rész azért is nagyon fontos, mert a korábbi Prolog verziók nem rendelkeztek objektum-orientált kiterjesztéssel, így a tanulókat nem volt könnyű motiválni.

Ebben a Prolog verzióban azonban már nagyon könnyen lehet objektum-orientált programokat készíteni. Ebben a fejezetben az elemek elektronszerkezete esettanulmányt objektum-orientált módon is elkészítjük a tanulókkal. [61]

A kurzus törzsanyaga tehát a következő leckeiből áll, melyek a programozott oktatás módszertanát követik:

- Programozás Prologban,
- Mintaillesztés és visszalépés,
- Fagyalt-vásárlás mintafeladat megoldása lépésekben,
- Rekurzív függvény-hívások Prologban,
- Vágás (cut) művelet,
- Tagadás,
- Ciklusszervezés rekurzióval,
- Nagycsalád (kibővített családfa) mintafeladat megoldása lépésekben,
- Listák és összetett kifejezések kezelése Prologban,
- Elemek elektronszerkezete esettanulmány,
- Objektum-orientált programok készítése WIN-Prologban,
- Elemek elektronszerkezete (objektum-orientált) esettanulmány.

Ezekben a lecekben ellenőrző kérdések, valamint elágazási pontok vannak elhelyezve. Amennyiben a tanuló nem tudja kiválasztani a helyes választ, abban az esetben egy részletesebb, kisebb lépésekben haladó útvonalra kerül.

A helyes válasz megadása után haladhat tovább a lecke feldolgozásával. A 25. ábra az „*Elemek elektronszerkezete esettanulmány*” lecke folyamatábrájából szemléltet néhány lépést. A 2.lépésnél a kérdésre adott válasznak megfelelően halad tovább a hallgató. Amennyiben helyesen válaszolt, haladhat tovább (az ábrán lefelé lép), míg ha hibás választ adott, akkor a hibaüzenet kijelzése után kap egy részletesebb magyarázatot (az ábrán a bal oldali ágra kerül), majd újra visszakérül a 2.lépéshez.



### V.3.1. Minta-adatbázis

Tekintsük a következő minta-adatbázist a megadott 3 táblával, a megadott kapcsolatokkal. Ebben az adatbázisban településekről tárolunk adatokat (településkód, településnév, terület (km<sup>2</sup>-ben), népesség, kistérségkód, oktatáskód). Minden települést egy kistérségbe sorolunk be, s minden településhez megadjuk az adott településen megvalósuló legmagasabb oktatási formát.

6. táblázat

település tábla

település-kód	településnév	terület	népesség	kistérségkód	oktatáskód
1	Eger	104,77	58976	k1	o4
2	Lőrinci	33,29	5968	k4	o3
3	Heves	99,21	11450	k5	o3
4	Bélapátfalva	36,17	3391	k1	o2
5	Mónosbél	14,24	399	k6	o1
6	Gyöngyös	54,1	33807	k3	o4
7	Abasár	26,65	3115	k3	o2
8	Füzesabony	46,34	7986	k2	o3
9	Aldebrő	21,78	791	k2	o2

Forrás: saját

kistérség tábla

7. táblázat

kistérségkód	kistérségnév
k1	egri
k2	füzesabonyi
k3	gyöngyösi
k4	hatvani
k5	hevesi
k6	pétervásárai

Forrás: saját

8. táblázat

oktatásfajta tábla (legmagasabb oktatásfajta az adott településen)

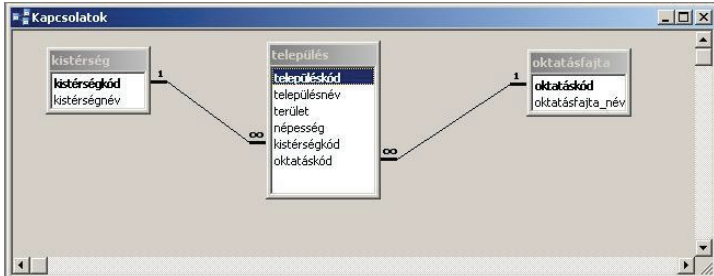
oktatáskód	oktatásfajta_név
o1	Óvoda
o2	Általános iskola
o3	Középiskola
o4	Felsőoktatási intézmény

Forrás: saját

A táblák közötti kapcsolatokat, valamint a táblák elsődleges kulcsait szemlélteti a 26. ábra.

26. ábra

Kapcsolatok a táblák között:



Forrás: saját

### V.3.2. A minta-adatbázis Prolog nyelven

Az előző pontban ismertetett minta-adatbázist Prolog tényekkel fogom megadni. Vegyük észre, hogy a tábla minden egyes sorának (rekordjának) megfeleltethetünk egy-egy Prolog tényt. Azokat a definíciókat, amelyek csak tényállításokból állnak a logikai programozás terén adatbázis definícióknak nevezik [23].

A három tábla rekordjainak megfelelő Prolog tények sémája így adható meg:

**település**(településkód, településnev, terület, népesség, kistérségkód, oktatáskód).

**kistérség**(kistérségkód, kistérségnev).

**oktatás**(oktatáskód, oktatásfajta\_név).

### **A Prolog tények a következők:**

település(1, 'Eger', 104.77, 58976, k1, o4).

település(2, 'Lőrinci', 33.29, 5968, k4, o3).

település(3, 'Heves', 99.21, 11450, k5, o3).

település(4, 'Bélapátfalva', 36.17, 3391, k1, o2).

település(5, 'Mónosbél', 14.24, 399, k6, o1).

település(6, 'Gyöngyös', 54.1, 33807, k3, o4).

település(7, 'Abasár', 26.65, 3115, k3, o2).

település(8, 'Füzesabony', 46.34, 7986, k2, o3).

település(9, 'Aldebrő', 21.78, 791, k2, o2).

kistérség(k1, egri).

kistérség(k2, füzesabonyi).

kistérség(k3, gyöngyösi).

kistérség(k4, hatvani).

kistérség(k5, hevesi).

kistérség(k6, pétervársárai).

oktatás(o1, 'Óvoda').

oktatás(o2, 'Általános iskola').

oktatás(o3, 'Középiskola').

oktatás(o4, 'Felsőoktatási intézmény').

### **V.3.3. Feladatok és megoldásuk SQL és Prolog nyelven**

Ebben a részben különböző feladatok (lekérdezések) megoldását adom meg SQL és Prolog nyelven, amellyel a nyelvek hasonló feladatmegoldási mechanizmusára hívom fel a figyelmet.

Emlékeztetőül szeretném megjegyezni, hogy a Prolog kifejezések argumentumaiban a változókat nagybetűvel kell kezdeni, míg a konstansokat kisbetűvel. A konstansok megadhatóak egyszeres aposztrófok között is.

**1. Melyek azok a települések, amelyek népessége 10 000 főnél nagyobb?**

**SQL:**

```
SELECT településkód, településnév  
FROM település  
WHERE népesség > 10000;
```

**Prolog:**

népes\_település(Településkód, Településnév) :- település(Településkód,  
Településnév, \_, Népesség, \_, \_), Népesség>10000.

**2. Mennyi az adott település népsűrűsége fő/km<sup>2</sup>-ben (népesség/terület)?**

**SQL:**

```
SELECT településkód, településnév, népesség/terület AS népsűrűség  
FROM település;
```

**Prolog:**

népsűrűség(Településkód, Településnév, Népsűrűség) :- település(Településkód,  
Településnév, Terület, Népesség, \_, \_), Népsűrűség is Népesség/Terület.

**3. Melyek azok a települések, amelyek népessége 5000 főnél nagyobb és az egri kistérséghez tartoznak?**

**SQL:**

```
SELECT település.településkód, település.településnév  
FROM település, kistérség  
WHERE (település.kistérségkód = kistérség.kistérségkód) AND  
((település.népesség > 5000) AND (kistérség.kistérségnév = "egri")) ;
```

**Prolog:**

egri\_kist\_nepes(Településkód, Településnév) :- település(Településkód,  
Településnév, \_, Néesség, Kistérségkód, \_), Néesség > 5000,  
kistérség(Kistérségkód, egri).

Összegzésként megállapítható, hogy a Prolog, illetve a logikai programozás, oktatásakor érdemes az adatbáziskezeléssel, illetve ezen belül az SQL nyelvvel a közös pontokat keresni. Ezáltal mind a logikai programozás, mind pedig az adatbáziskezelés sokszor bonyolultnak tűnő összefüggéseit tudjuk alaposabban elsajátíttatni tanulóinkkal.



## VI. A kísérleti kurzusok eredményeinek, tapasztalatainak kiértékelése, elemzése

### VI.1. Bevezetés

2011. szeptember 5. és november 18. között összetett kétcsoportos pedagógiai kísérletet végeztem 3 gyöngyösi oktatási intézmény közreműködésével (Vak Bottyán János Katolikus Műszaki és Közgazdasági Középiskola, József Attila Szakközépiskola, Szakiskola és Kollégium, Károly Róbert Főiskola). Mindegyik intézményben volt egy kísérleti csoport, valamint egy kontroll csoport.

A 9.táblázat mutatja a kísérletben résztvevő csoport-létszámokat.

**9. táblázat**

<b>Helyszín</b>	<b>Kísérleti csoport (fő)</b>	<b>Kontroll csoport (fő)</b>	<b>Összesen (fő)</b>
Vak Bottyán János Katolikus Műszaki és Közgazdasági Középiskola	26	35	<b>61</b>
József Attila Szakközépiskola, Szakiskola és Kollégium	23	38	<b>61</b>
Károly Róbert Főiskola	29	38	<b>67</b>
<b>Összesen (fő):</b>	<b>78</b>	<b>111</b>	<b>189</b>

Forrás: saját

A Vak Bottyán János Katolikus Műszaki és Közgazdasági Középiskola intézményében vizsgálataimban 2 osztály volt segítségemre (12.b, 13.c).

A kísérleti csoportot kötelező jelleggel a 13.c osztály alkotta.

A kísérleti csoport létszáma 26 fő volt. Az átlagéletkor 18,6 év, míg a csoportnak az előző évi, év végi matematika jegy átlaga 3,6 volt.

A kontroll csoport létszáma 35 fő volt. Az átlagéletkor 17,5 év, míg a csoportnak az előző évi, év végi matematika jegy átlaga 3,7 volt.

A József Attila Szakközépiskola, Szakiskola és Kollégium intézményében 3 osztályt vontam be a kísérletbe (11.a, 12. a, 12.b). A kísérleti csoportot kötelező jelleggel a 12.a osztály alkotta, valamint néhány 11.a-s tanuló (4 fő), akik érdeklődtek a téma iránt és önként vállalták a csoportban való részvételt.

A kísérleti csoport létszáma 23 fő volt. Az átlagéletkor 18,0 év, míg a csoportnak az előző évi, év végi matematika jegy átlaga 2,7 volt.

A kontroll csoport létszáma 38 fő volt. Az átlagéletkor 17,6 év, míg a csoportnak az előző évi, év végi matematika jegy átlaga 2,6 volt.

A Károly Róbert Főiskolán vizsgálataimba az első éves gazdaságinformatikus BSc-s hallgatók kerültek bevonásra. A kísérleti csoportban való tagságot önkéntes alapon lehetett választani.

A kísérleti csoport létszáma 29 fő volt. Az átlagéletkor 19,5 év, míg a csoportnak az előző évi, év végi matematika jegy átlaga 3,3 volt.

A kontroll csoport létszáma 38 fő volt. Az átlagéletkor 19,8 év, míg a csoportnak az előző évi, év végi matematika jegy átlaga 3,2 volt.

Megállapítható tehát, hogy a 3 helyszínen voltak kismértékű eltérések a tanulói csoportok között. Az összetett kétcsoportos pedagógiai kísérletet ezért is tartottam jó megoldásnak, ugyanis itt mindegyik helyszínen a kontroll csoportok kontrollálni tudták a kísérlet hatását.

A kísérleti csoportokat heti két órában oktattam 11 héten keresztül. A kurzusokat e-learning rendszerrel támogattam, a <http://patronus.karolyrobert.hu> portálon elérhetővé tettem minden kísérleti csoportbeli tag számára a Logikai programozás kurzust. [62]

A kurzus elején és a kurzus végén problémamegoldó feladatsorokat (bemeneti és kimeneti teszt) oldattam meg mind a három helyszínen, mindegyik csoporttal. A tesztfeladatok összeállításánál Róka Sándor feladatgyűjteményeit, a Zrínyi Ilona Matematikaverseny feladatait, valamint a 2003-as OECD-PISA Nemzetközi tanulói teljesítménymérés problémamegoldó feladatait használtam fel. [63], [64], [65], [66], [67], [68], [69], [70], [71]

A bemeneti és a kimeneti tesztek eredményeit statisztikailag elemeztem SPSS programcsomag segítségével. [72], [73]

A kísérlet sikeres lebonyolítása alapján fogalmaztam meg tézisemet, miszerint:  
**A problémamegoldó gondolkodás hatékonyan fejleszthető a logikai programozás oktatásával.**

## **VI.2. A Logikai programozás kísérleti kurzus tananyagának bemutatása**

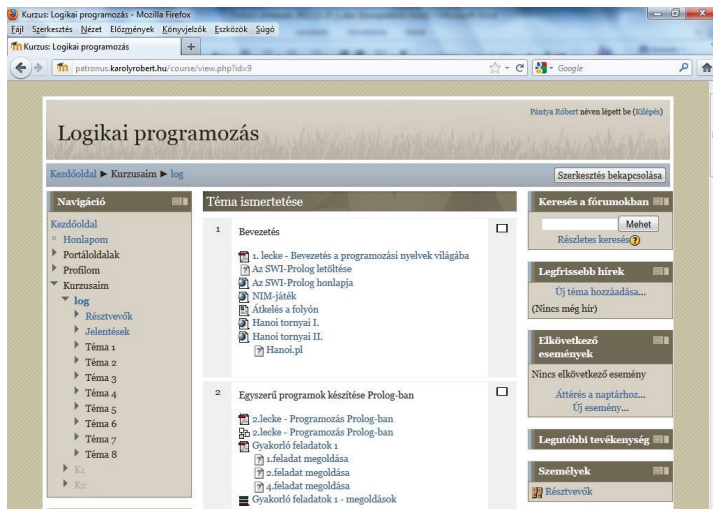
Ebben a fejezetben részletesen bemutatom a kísérleti kurzus tananyagát. A Logikai programozás kurzus a <http://patronus.karolyrobert.hu> portálon Moodle 2.0-ás környezetben került megvalósításra. Ez a szoftver verzió azért is szerencsés, mert ebben a Moodle *Conditional Activities* szolgáltatásának lehetőségei miatt a programozott oktatás olyan segítséget kap, hogy azt is lehet irányítani, hogy milyen sorrendben jelenjenek meg az egyes tananyag-elemek.

A kísérlet 11 hetének tananyaga 7 fejezetre tagolódt. A szemléltetéshez Flash-animációkat készítettem (összesen 35 darabot). Programozott oktatással leckéket illesztettem be, melyek a tananyagot magyarázták el. Három tudásellenőrző teszt is megvalósításra és felhasználásra került, több mint 300 kérdést tartalmazó kérdésbank felhasználásával.

A kísérleti csoport tagjainak az e-learning rendszerben hozzáférést biztosítottam.

A sikeres belépés utáni képernyő látható a következő ábrán.

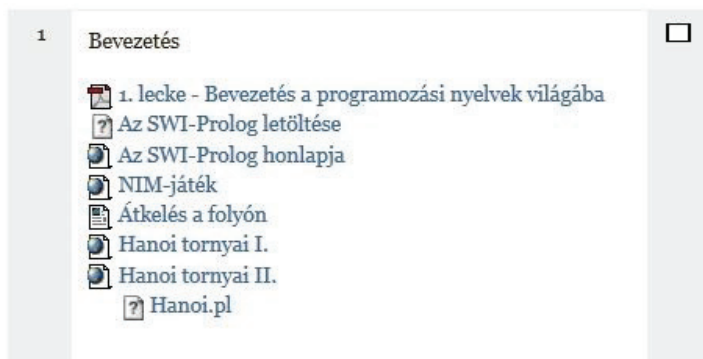
27. ábra



Forrás: saját

Az 1. fejezet a kurzus bevezetését tartalmazta. A következő ábrán látható a fejezet megjelenési formája.

28. ábra



Forrás: saját

Ebben a fejezetben áttekintésre kerültek a programozási nyelvek legfontosabb jellemzői, csoportosítási lehetőségei (amatőr – professzionális, alacsonyszintű – magasszintű, imperatív – deklaratív, számítási modellek alapján). Bemutatásra kerültek a programozási nyelvek generációi is. Végül a tanításra kerülő Prolog nyelv került besorolásra a csoportosításoknak megfelelően.

Ebben a fejezetben megadtam a telepítendő Prolog nyelv verziójának a letölthetőségét tartalmazó web-oldalt. [74]

Az SWI-Prolog mellett döntöttem, mivel ez ingyenesen használható, valamint nemcsak Windows-hoz létezik elérhető verzió, hanem Linux és MacOS-hez is.

Nagyon népszerűek voltak és a motivációban nagyon sokat segítettek az online logikai játékok (NIM-játék [75], Hanoi tornyai [58], [59], Átkelés a folyón [76]). A NIM-játék azért is nagyon szerencsés, mivel a nyerő stratégia ismerete nélkül nem lehet megverni ezt az ellenfelet. A nyerő stratégia megtanítása után, azonban már mindenkinek nagyon sok sikerélménye volt, emellett betekintést is nyert a játékelméletbe. [1]

Az 1. fejezet feladata volt az e-learning felület kezelésének a megtanítása is. A két kísérleti középiskolában a kísérlet lefolytatása idején még nem működött e-learning rendszer.

Úgy gondolom, hogy hamarosan üzemeltetni fognak egy ilyen rendszert, mivel nagyon szívesen használták ezt a felületet a hallgatók. A napló fájlok alapos böngészése során tapasztaltam, hogy volt amikor reggel 5 órakor is történt belépés, illetve éjjel után is dolgoztak többször a tanulók. Az e-learning rendszerek egyik nagy előnye pontosan ez az időbeli (és térbeli) rugalmasság, amely megengedi a tanulónak a személyéhez legjobban igazodó tanulási stílus alkalmazását.

A 2. fejezetben elkezdődött a Prolog-ban való programozás. A „*Gyakorló feladatsor 1*” a **C függelékben** megtalálható. A fejezet célja az egyszerű Prolog szintaktika megtanítása volt. Ez a feladatsor úgy vélem, hogy nem okozott gondot egyik intézményben sem.

A legfontosabb logikai műveleteket (ÉS, VAGY, KIZÁRÓ VAGY, NOT, IMPLIKÁCIÓ, EKVIVALENCIA) flash-animációkkal szemléltettem.

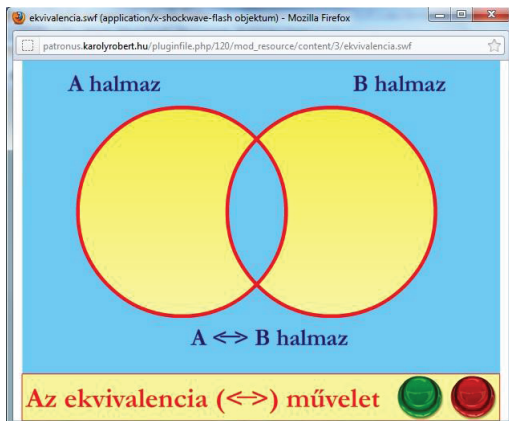
A következő ábrákon látható a fejezet megjelenési formája, valamint egy képernyőkép az *ekvivalencia* logikai művelet animációjáról.

29. ábra



Forrás: saját

30. ábra



Forrás: saját

A Start nyomógommbal (zöld gomb) lehetett elindítani az animációt és a Stop gommbal (piros gomb) bármikor meg lehetett állítani. Ezzel minden tanuló saját magának tudta ütemezni az animáció lefutását.

A tanulók otthoni munkavégzését elősegítendő házi feladatot kellett megoldaniuk és határidőre az adott linkhez feltölteniük. Bár minden csoportban nagyon sokan elkészítették a feladatot, de mindegyik csoportomban voltak olyan tanulók, akiknek ez nem sikerült. Az e-learning rendszer előnyeik közt kell megemlíteni, hogy pontosan rögzítette, hogy kik, mikor készítették el a feladatot, s így a tanulói aktivitást is lehetett mérni segítségével.

A 3. fejezetben a Prolog lényegi működési mechanizmusával és vezérlési stratégiájával, a mintaillesztéssel és visszalépéssel, ismerkedhetek meg tanulóim. A „*Gyakorló feladatok 2*” ugyancsak a **C függelékben** található. A fejezetet itt is házi feladat elkészítése és a rendszerbe való feltöltése zárta.

31. ábra



Forrás: saját

Az *Autók*, a *Kedvenc állatok* és a *Játékok* feladatokhoz szemléltető animációk készültek. A tanulóknak a feladatok forrásprogramját elérhetővé tettem a felületen.

32. ábra

### Autók feladat



Forrás: saját

Ez az animáció halmazokba sorolja az adott szempontoknak megfelelő autókat, mellyel nagyon egyszerűen szemlélteti a logikai ÉS műveletet.

#### A feladat forrásprogramja:

```
robinak_tetszik(ford_t).
robinak_tetszik(ferrari).
robinak_tetszik(skoda_fabia).
robinak_tetszik(nissan_note).
jutkanak_tetszik(ferrari).
jutkanak_tetszik(nissan_note).
```



jutkának\_tetszik(toyota\_hibrid).

jutkának\_tetszik(smart).

van\_rá\_pénz(nissan\_note).

van\_rá\_pénz(skoda\_fabia).

van\_rá\_pénz(smart).

van\_rá\_pénz(trabant).

van\_rá\_pénz(wartburg).

A **vesznek\_autót** szabályt, és az ezzel kapcsolatos célállítás megoldásának mechanizmusát szemlélteti a következő ábra. Véleményem szerint itt látszik meg az igazi „ereje” egy ilyen oktatási segédletnek (flash, e-learning).

**vesznek\_autót(X) :- robinak\_tetszik(X), jutkának\_tetszik(X), van\_rá\_pénz(X).**

33. ábra

Autok\_1\_4.swf (application/x-shockwave-flash objektum) - Mozilla Firefox

patonus.karolyrobert.hu/pluginfile.php/160/mod\_resource/content/1/Autok\_1\_4.swf

robinak\_tetszik(ford\_t).  
robinak\_tetszik(ferrari).  
robinak\_tetszik(skoda\_fabia).  
robinak\_tetszik(nissan\_note).  
jutkának\_tetszik(ferrari).  
jutkának\_tetszik(nissan\_note).  
jutkának\_tetszik(toyota\_hibrid).  
jutkának\_tetszik(smart).  
van\_rá\_pénz(nissan\_note).  
van\_rá\_pénz(skoda\_fabia).  
van\_rá\_pénz(smart).  
van\_rá\_pénz(trabant).  
van\_rá\_pénz(wartburg).

**Megoldás!**  
X = nissan\_note

vesznek\_autót(X) :- robinak\_tetszik(X),  
                          jutkának\_tetszik(X), van\_rá\_pénz(X).

| ?- vesznek\_autót(X).  
X = nissan\_note

Készítette: Pántya Róbert

Milyen autót vegyenek?

Forrás: saját

## Kedvenc állatok feladat



Forrás: saját

Ebben a feladatban lyukkártyákkal szemléltettem azt, hogy melyik személy milyen kedvenc állatot tartott már.

**A feladat forrásprogramja:**

tartott('Róbi', 'kutya').

tartott('Róbi', 'macska').

tartott('Róbi', 'aranyhal').

tartott('Réka', 'teknős').

tartott('Réka', 'kutya').

tartott('Réka', 'macska').

tartott('Réka', 'aranyhal').

tartott('Nóra', 'kutya').

tartott('Nóra', 'macska').

tartott('Nóra', 'aranyhal').

tartott('Nóra', 'tengerimalac').

tartott('Jutka', 'kutya').  
 tartott('Jutka', 'macska').  
 tartott('Jutka', 'papagáj').  
 tartott('Jutka', 'pinty').  
 tartott('Jutka', 'tengerimalac').  
 tartott('Ádám', 'teknős').  
 tartott('Ádám', 'kutya').  
 tartott('Ádám', 'macska').  
 tartott('Ádám', 'aranyhal').  
 tartott('Ádám', 'papagáj').  
 tartott('Ádám', 'pinty').  
 tartott('Ádám', 'tengerimalac').

Ha kíváncsiak vagyunk arra, hogy melyik állatot tartotta már mindenki, akkor a **mindenki\_tartott** szabállyal, valamint az általa megfogalmazott célállítással kaphatunk rá választ. A lyukkártyák egymásra helyezését, mint megoldást, a 35. ábra szemlélteti.

**mindenki\_tartott(X) :-** tartott('Robi', X), tartott('Réka', X), tartott('Nóra', X), tartott('Jutka', X), tartott('Ádám', X).

35. ábra



Forrás: saját

## Játékok feladat

Jutka		✓				✓	✓
Réka	✓	✓	✓	✓	✓	✓	✓
Nóra		✓			✓		✓
Robi	✓	✓	✓	✓	✓	✓	

szeret('Jutka', 'labdázni').  
 szeret('Jutka', 'tollasozni').  
 szeret('Jutka', 'ugrókötelezni').  
 szeret('Réka', 'gombfocizni').  
 szeret('Réka', 'labdázni').  
 szeret('Réka', 'kártyázni').  
 szeret('Réka', 'sakkozni').  
 szeret('Réka', 'társasjátékozni').  
 szeret('Réka', 'tollasozni').  
 szeret('Réka', 'ugrókötelezni').

szeret('Nóra', 'labdázni').  
 szeret('Nóra', 'társasjátékozni').  
 szeret('Nóra', 'ugrókötelezni').  
 szeret('Robi', 'gombfocizni').  
 szeret('Robi', 'labdázni').  
 szeret('Robi', 'kártyázni').  
 szeret('Robi', 'sakkozni').  
 szeret('Robi', 'társasjátékozni').  
 szeret('Robi', 'tollasozni').

Készítette: Pántya Róbert

**Ki, mit szeret játszani ?**

Forrás: saját

A **Játékok** feladatban az egyes személyek kedvenc játékait, szórakozását gyűjthetjük ki.

A 4. fejezetben a „Gyakorló feladatsor 3” kapott helyet (ez ugyancsak a **C függelékben** megtalálható), melynek a segítségével a mintaillesztést és a visszalépéses feladatmegoldó algoritmust tudtam alaposan begyakoroltatni. Ebben a fejezetben ismerték meg a tanulók a **Rákóczi család** feladatot is, melyre a 6. fejezetben is támaszkodtam, a rekurzió tanításánál.

A 4. fejezet megjelenési formája látható a következő ábrán.

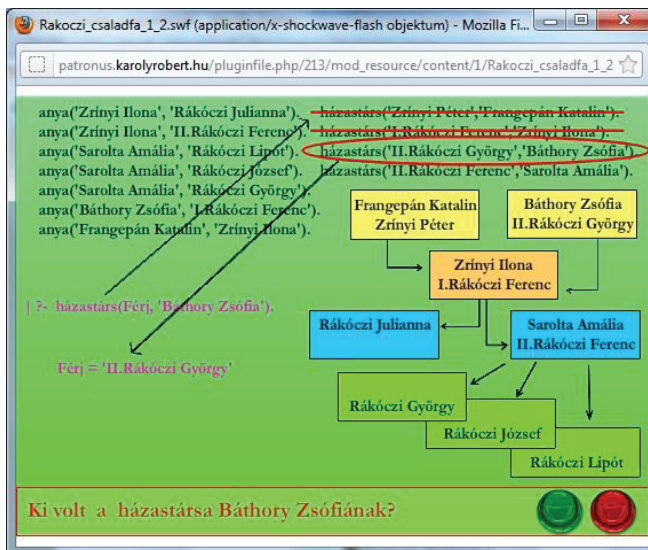
37. ábra



Forrás: saját

A *Rákóczi család* feladathoz 5 animációt is készítettem, melyet nagyon jól értékelték a tanulók. Egy ilyen bonyolultabb feladat szemléltetéséhez (különösen e-learning rendszerben) mindenképpen szükség van animációkra.

38. ábra



Forrás: saját

Az 5. fejezetben elhelyezésre került egy olyan logikai gyakorló feladatsor („*Gyakorló feladatok 4*” – *C függelék*), melyben mindenki megszodálhatta a Prolog, s így a logikai programozás, igazi erejét. Ebben a részben a már klasszikussá vált logikai feladatokat (*Portia ládáit, Ki nem mondott igazat?* stb.) oldottunk meg néhány Prolog program-sor segítségével. Ezeknek a feladatoknak a megoldása nagyon népszerű volt az egyes csoportokban).

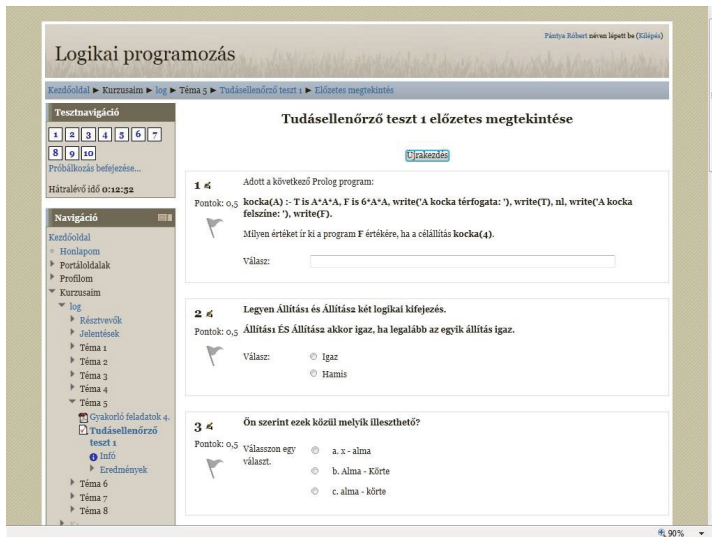
### 39. ábra



Forrás: saját

Ebbe a fejezetbe került beillesztésre az első tudásellenőrző teszt. Ebben több mint 100 kérdésből véletlenszerűen választott ki a rendszer 10 tesztkérdést, melyek mindegyikére, jó megoldás esetén, 0,5 pont volt kapható. A tanulók korlátlan számban kitölthették ezeket a teszteket (természetesen minden újabb kitöltéskor véletlenszerűen, újabb 10 tesztkérdést kaptak). A kurzus végén osztályzatot is kaptak a tanulók, melynek a meghatározásába ezeket a megszerzett pontokat is beszámítottam. A korlátlan próbálkozás megengedésének a nyilvánvaló célja az volt, hogy minél többször töltsék ki ezeket a teszteket, ezáltal minél jobban gyakorolják be, tanulják meg ezeket az ismereteket.

A következő ábra bemutatja a teszt egy részletét, ahogyan az a hallgatóknak megjelent.



Forrás: saját

A Moodle e-learning rendszer lehetővé teszi számos statisztika elkészítését is a tesztekkel kapcsolatban. Ez nagyban elő tudja segíteni a tanár munkáját. A következő képről (41. ábra) a **Tudásellenőrző teszt 1** legfontosabb statisztikai jellemzői olvashatóak le. Ez megadja például azt, hogy összesen 68-an próbálták tudásukat felmérni ennek a segítségével. Összesen 251 próbálkozás történt a tesztek megoldására. Az első próbálkozások átlagos eredménye 51 %, míg az összes próbálkozás átlagos eredménye 65 % volt.

*A Tudásellenőrző teszt 1***Teszt adatai**

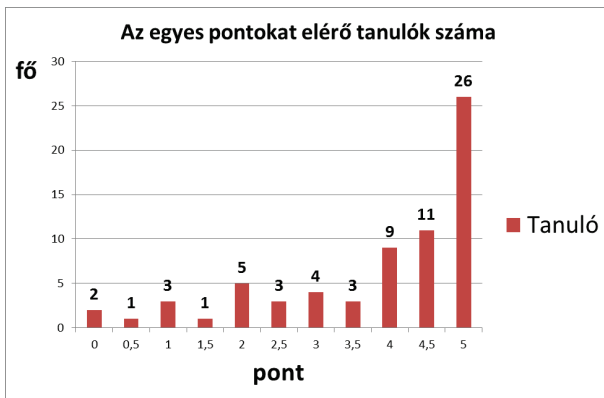
<div> <div>A teljes jelentés letöltése mint</div> <div>vesszővel elválasztott értékeket tartalmazó szöveges állományként ▼</div> </div>	
Teszt neve	Tudásellenőrző teszt 1
Kurzus neve	Logikai programozás
Első próbálkozások száma	68
Összes próbálkozások száma	251
Az első próbálkozások átlagos pontszáma	51,0%
Az összes próbálkozás átlagos pontszáma	65,0%
Medián pont (első próbálkozások esetén)	55,0%
Szórás (első próbálkozások esetén)	31,8%
Pontszámeloszláshoz tartozó aszimmetria (első próbálkozások esetén)	-0.11388218088158
Pontszámeloszláshoz tartozó csúcsosság (első próbálkozások esetén)	-1.2332152049513
Belső konzisztencia együtthatója (első próbálkozások esetén)	83,5%
Hibaarány (első próbálkozások esetén)	40,6%
Standard hiba (első próbálkozások esetén)	12,9%

Forrás: saját

A következő ábra azt mutatja meg, hogy a különböző pontszám-kategóriákat (0-tól 5-ig) hány tanuló érte el. Itt az egyes tanulóknak csak a legmagasabb pontszáma került ábrázolásra. Erről az ábráról leolvasható, hogy azon 68 tanuló közül, aki megpróbálkozott a teszt kitöltésével, 26 jutott el a maximális pontszámig.



42. ábra



Forrás: saját

A 6. fejezetben került ismertetésre a rekurzió, amely nem sorolható a könnyen, egyszerűen tanítható/tanulható tananyag-részek közé. A jelentősége azonban nagy, mivel a ciklusszervezés a rekurzió segítségével valósítható meg a deklaratív programozásban. Ezért ehhez a feladatrészhöz különösen sok animációt, szemléltető példát készítettem.

A „Gyakorló feladatok 5” (*C függelék*) a *Mappák* feladatot, az *Árpád-házi uralkodók*, valamint a *Rákóczi család* feladatokat tartalmazza.



Forrás: saját

A **Mappák** feladatot választottam a rekurzió bevezető példájának. Azért esett erre a választásom, mivel ennek „elméleti” háttérét mindenki ismeri. A **tartalmaz** tényállítás megmutatja, hogy melyik mappának melyik a közvetlen őse, illetve leszármazottja. A **bennevan** szabály pedig megadja azt, hogy mikor tekinthető egy mappa egy másik mappa almappájának.

A 44. ábrán látható a **tartalmaz** tényállítások, valamint a **bennevan** szabályok pontos megfogalmazása.

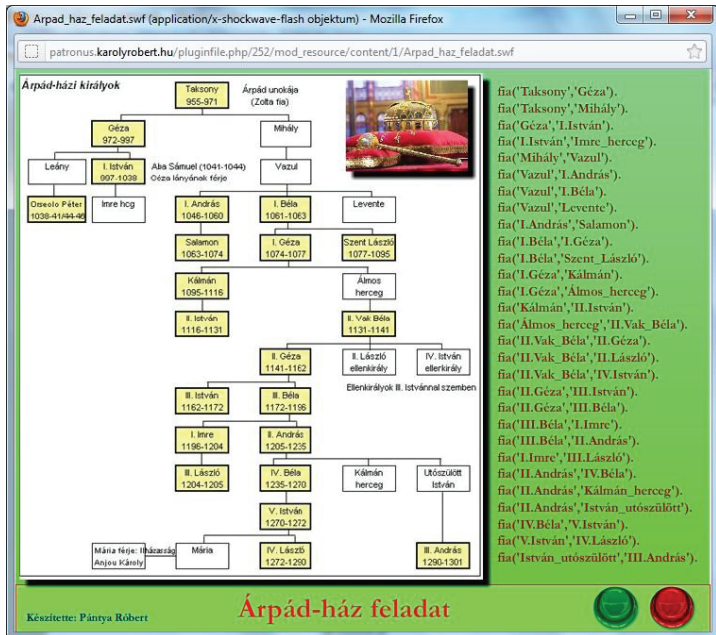
## Mappák feladat



Forrás: saját

A **Mappák** feladat után egy kis módosítással juthatunk az *Árpád-házi uralkodók* feladathoz. Ebben a feladatban az egyszerűség kedvéért csak az apai ágat tartottuk nyilván. Az előző feladat **tartalmaz** tényét a **fia** tényállításra cseréltük, míg a **bennevan** szabály helyett az **utód** szabályt alkottuk meg. Ennél a feladatnál már érdekesebb szabályokat is el lehetett készíttetni a tanulókkal (testvér, unokatestvér, nagypapa, dédapapa, stb.)

## Árpád-házi uralkodók feladat



Forrás: saját

Ennek a fejezetnek méltó lezárása volt a **Rákóczi család** feladat, ahol már nemcsak az apai, hanem az anyai ági leszármazást is vizsgáltuk, és még további családi viszonyokat (apai nagyapa, anyai nagyapa, apai nagyanya, anyai nagyanya, anyós, após, meny, vő, stb.)

### Néhány szabály a családi viszonyokra:

apa(Apu, Gyerek) :- házastárs(Apu, Anyu), anya(Anyu, Gyerek).

utód(X, Ős) :- anya(Ős, X).

utód(X, Ős) :- anya(Valaki, X), utód(Valaki, Ős).

utód2(X, Ős) :- apa(Ős, X).

utód2(X, Ős) :- apa(Valaki, X), utód2(Valaki, Ős).

anyós(Meny, Anyós) :- házastárs(X, Meny), anya(Anyós, X).

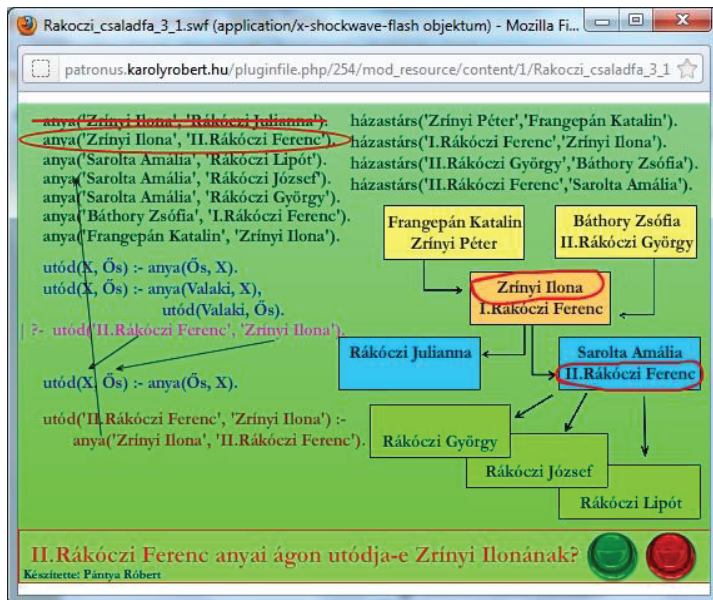
anyós(Vő, Anyós) :- házastárs(Vő, X), anya(Anyós, X).

após(Meny, Após) :- házastárs(X, Meny), apa(Após, X).

após(Vő, Após) :- házastárs(Vő, X), apa(Após, X).

46. ábra

### Rákóczi család feladat



Forrás: saját

A fejezetet a *Tudásellenőrző teszt 2* zárta, melynek statisztikáit tartalmazó képernyőképét, grafikonját mutatja a következő két ábra.

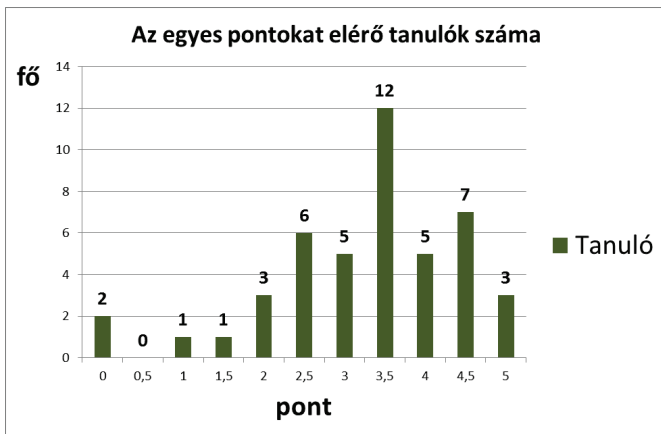
## A Tudásellenőrző teszt 2

## Teszt adatai

A teljes jelentés letöltése mint vesszővel elválasztott értékeket tartalmazó szöveges állományként ▼

Teszt neve	Tudásellenőrző teszt 2
Kurzus neve	Logikai programozás
Első próbálkozások száma	45
Összes próbálkozások száma	214
Az első próbálkozások átlagos pontszáma	39,6%
Az összes próbálkozás átlagos pontszáma	48,4%
Medián pont (első próbálkozások esetén)	40,0%
Szórás (első próbálkozások esetén)	26,2%
Pontszámeloszláshoz tartozó aszimmetria (első próbálkozások esetén)	0.24117938083205
Pontszámeloszláshoz tartozó csúcsosság (első próbálkozások esetén)	-0.70912080908116
Belső konzisztencia együtthatója (első próbálkozások esetén)	72,2%
Hibaarány (első próbálkozások esetén)	52,8%
Standard hiba (első próbálkozások esetén)	13,8%

Forrás: saját



Forrás: saját

A 7. és egyben záró fejezetben a rekurzió került alapos gyakoroltatásra, valamint a listákkal, összetett adatszerkezetekkel ismertetem meg a tanulókat. A „*Gyakorló feladatok 6*” a **C függelékben** található. Bőséggel készültek szemléltető animációk ehhez a fejezethez is, melyek közül néhány képernyőkép kiemelésre került a következő ábrán.

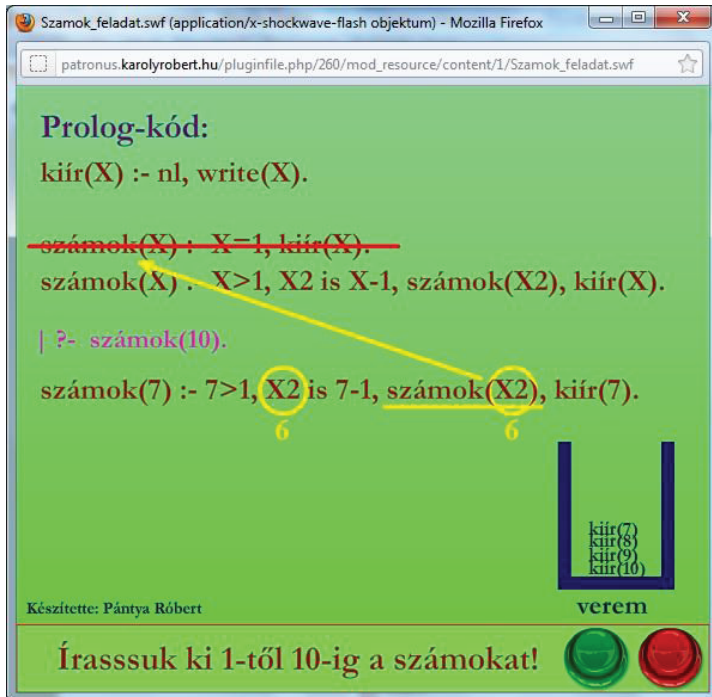
49. ábra



Forrás: saját

Ebben a záró fejezetben már igazán érdekes problémák megoldására is képesek voltak tanulóim. A **Számok** feladat segítségével (ahol megadott egész számig írjuk ki az egész számokat) a jobbrekurzió, balrekurzió közötti különbségre is fel tudtam hívni a figyelmüket. Ezzel egyidőben a **verem** adatszerkezet is szemléltethetővé vált.

## Számok feladat



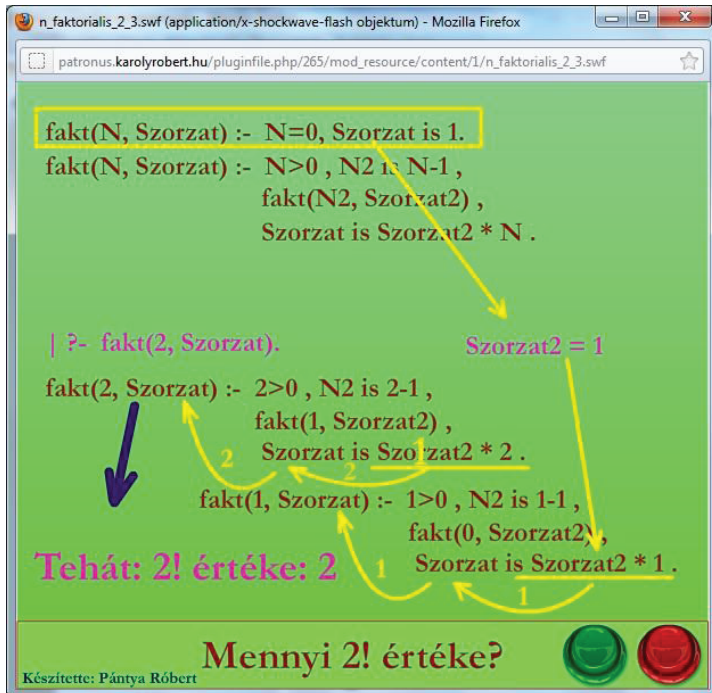
Forrás: saját

Az **n faktoriális (n!)** feladat is nagyon hasznos a rekurzió tanításakor.

A 0!, 1!, 2! kiszámolásának mechanizmusát ugyancsak animációkkal szemléltettem, melyek közül a 2! kiszámítását mutatja be a következő ábra.



## n! feladat

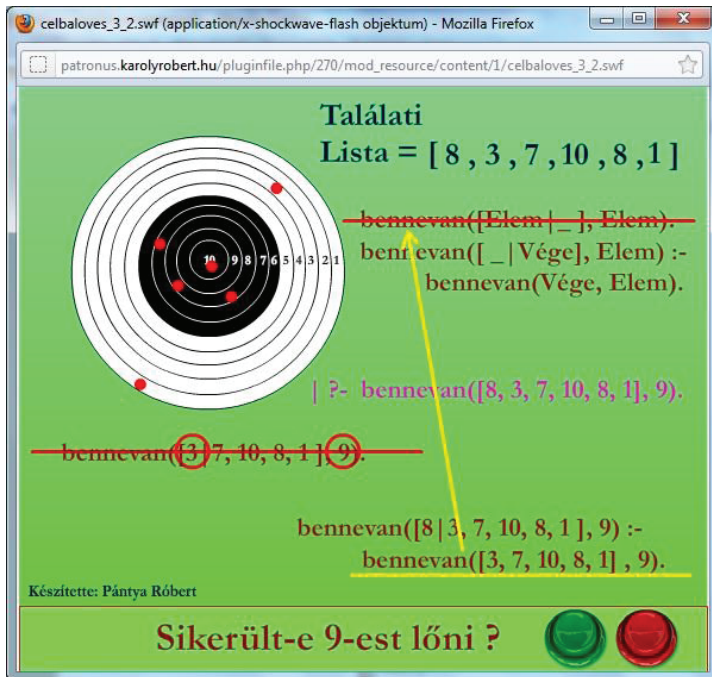


Forrás: saját

A *Célbalövés* és *Kártyák* feladatok pedig a listák kezelésével (Találati Lista [8, 3, 7, 10, 8, 1]), valamint az összetett adatszerkezetekkel (lap(t,10)) ismertette meg a tanulókat.

A fejezet végére képessé váltak a tanulók a kurzus elején elhelyezett *Hanoi tornyai* feladat megoldására is.

## Célbalövés feladat



Forrás: saját

## Kártyák feladat



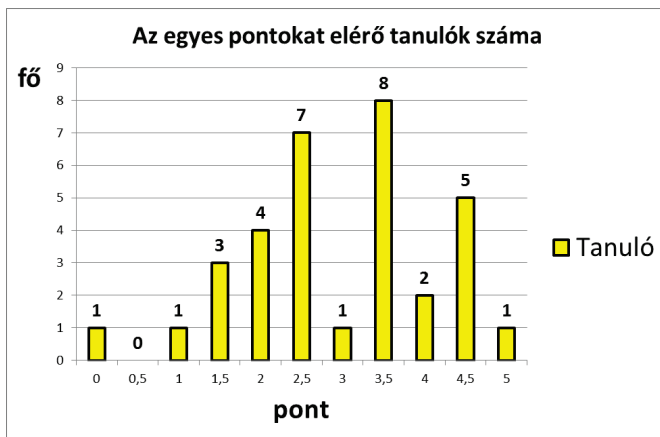
Forrás: saját

A fejezetet a *Tudásellenőrző* *teszt* 3 zárta, melynek statisztikáit tartalmazó képernyőképét, grafikonját mutatja a következő két ábra.

## A Tudásellenőrző teszt 3

## Teszt adatai

A teljes jelentés letöltése mint vesszővel elválasztott értékeket tartalmazó szöveges állományként	
Teszt neve	Tudásellenőrző teszt 3.
Kurzus neve	Logikai programozás
Első próbálkozások száma	33
Összes próbálkozások száma	151
Az első próbálkozások átlagos pontszáma	34,5%
Az összes próbálkozás átlagos pontszáma	47,6%
Medián pont (első próbálkozások esetén)	30,0%
Szórás (első próbálkozások esetén)	26,7%
Pontszámeloszláshoz tartozó aszimmetria (első próbálkozások esetén)	0.28432178532122
Pontszámeloszláshoz tartozó csúcsosság (első próbálkozások esetén)	-0.72079155732804
Belső konzisztencia együtthatója (első próbálkozások esetén)	75,6%
Hibaaarány (első próbálkozások esetén)	49,4%
Standard hiba (első próbálkozások esetén)	13,2%



Forrás: saját

### **VI.3. A Logikai programozás kísérleti kurzus eredményeinek statisztikai elemzése**

#### **VI.3.1. Módszer**

A kurzus elején és a kurzus végén problémamegoldó feladatsorokat (bemeneti és kimeneti teszt) oldattam meg mind a három helyszínen, mindegyik csoporttal.

A bemeneti teszt feladatai a D függelékben, míg az egyes helyszíneken az egyes csoportok tanulói által elért eredmények az E függelékben láthatóak. A tanulók nevét a táblázatban nem közöltem a személyes adatok védelme miatt.

A kimeneti teszt feladatai a F függelékben, míg az egyes helyszíneken az egyes csoportok tanulói által elért eredmények a G függelékben láthatóak.

Mind a bemeneti, mind pedig a kimeneti teszt 100 pontos volt. A csoportok azonosítását a 0 és az 1-es számjegyekkel végeztem. Minden helyszínen a 0 a kontroll csoportot, míg az 1 a kísérleti csoportot jelentette.

A bemeneti és a kimeneti tesztek eredményeit statisztikailag elemeztem SPSS programcsomag segítségével. Vizsgáltam azt, hogy van-e szignifikáns különbség a két csoport bemeneti illetve kimeneti tesztjének eredménye között.

A bemeneti, valamint a kimeneti tesztek eredményeit összehasonlítottam helyszínenként is, valamint az összes helyszínt egybevéve is, csoportonként.

A vizsgálatokat az adatok normalitás-vizsgálatával kezdtem, ugyanis az alkalmazható statisztikai módszert meghatározza az, hogy a mért adatok normális eloszlásúak-e.

A normalitás-vizsgálatot Kolmogorov-Smirnov, illetve Shapiro-Wilk módszerrel végeztem el.

Azokban az esetekben, amikor normál eloszlást tapasztaltam az eredményekben, akkor az F-próba elvégzése után lehetett dönteni, hogy a 2-mintás T-próba, vagy a Welch-próba alkalmazható a kísérleti és a kontroll csoportok eredményeinek az összehasonlítására.

Voltak olyan mérések is, ahol néhány kiugró adat miatt nem volt normál eloszlású a minta. Ezekben az esetekben a két minta összehasonlítását a Mann-Whitney nem-paraméteres (rangsorolósos) vizsgálattal végeztem el.

Az alábbi ábra az egyes feladatok kódjait tartalmazza:

**56. ábra**

Bemeneti teszt													
Ssz.:	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	össz.
Kód	be 1	be 2	be 3	be 4	be 5	be 6	be 7	be 8	be 9	be 10	be 11	be 12	be szum
Kimeneti teszt													
Ssz.:	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	össz.
Kód	ki 1	ki 2	ki 3	ki 4	ki 5	ki 6	ki 7	ki 8	ki 9	ki 10	ki 11	ki 12	ki szum

Forrás: saját

## VI.3.2. Bemeneti tesztek értékelése

### VI.3.2.1. József Attila Szakközépiskola, Szakiskola és Kollégium bemeneti tesztjei

A következő 2 táblázatban a kontroll és a kísérleti csoport bemeneti tesztjének leíró statisztikái láthatóak.

#### Kontroll csoport - József Attila Szakközépiskola, Szakiskola és Kollégium

Descriptive Statistics

**10. táblázat**

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
be_szum	38	60	9	69	28,84	11,746	137,974
Valid N (listwise)	38						

#### Kísérleti csoport - József Attila Szakközépiskola, Szakiskola és Kollégium

Descriptive Statistics

**11. táblázat**

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
be_szum	23	48	11	59	31,43	12,380	153,257
Valid N (listwise)	23						

Mivel a mért eredmények nem voltak egyiknél sem normál eloszlásúak, ezért a két minta összehasonlítását a Mann-Whitney nem-paraméteres (rangsorolós) vizsgálatlal hasonlítottam össze.

### Mann-Whitney Test

#### Ranks

12. táblázat

	csoport	N	Mean Rank	Sum of Ranks
be_szum	0	38	29,49	1120,50
	1	23	33,50	770,50
	Total	61		

#### Test Statistics(a)

13. táblázat

	be_szum
Mann-Whitney U	379,500
Wilcoxon W	1120,500
Z	-,857
Asymp. Sig. (2-tailed)	,391

a. Grouping Variable: csoport

Ebből megállapítható, hogy a két minta a kísérlet kezdetekor homogénnek tekinthető 5 % szignifikancia szinten ( $p=0,05$  hibaszinten), vagyis nincs szignifikáns különbség a két csoport bemeneti tesztje között.

### VI.3.2.2. Vak Bottyán János Katolikus Műszaki és

### Közgazdasági Középiskola bemeneti tesztje

A következő 2 táblázatban a kontroll és a kísérleti csoport bemeneti tesztjének leíró statisztikái láthatóak.

#### Kontroll csoport - Vak Bottyán János Katolikus Műszaki és Közgazdasági

#### Középiskola

#### Descriptive Statistics

14. táblázat

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
be_szum	35	63	25	88	47,91	12,387	153,434
Valid N (listwise)	35						

**Kísérleti csoport - Vak Bottyán János Katolikus Műszaki és Közgazdasági  
Középiskola**

**Descriptive Statistics**

**15. táblázat**

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
be_szum	25	50	21	71	52,28	13,378	178,960
Valid N (listwise)	25						

Mivel mindkét minta normál eloszlásúnak bizonyult, valamint az F-próba is alkalmazhatóvá tette, ezért a két minta összehasonlítását 2-mintás T-próbával végeztem el.

**Group Statistics**

**16. táblázat**

	csoport	N	Mean	Std. Deviation	Std. Error Mean
be_szum	0	35	47,91	12,387	2,094
	1	25	52,28	13,378	2,676

**17. táblázat**

**Independent Samples Test**

		Levene's Test for Equality of Variances		t-test for Equality of Means						
		F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference	
									Lower	Upper
be_szum	Equal variances assumed	,891	,349	-1,302	58	,198	-4,366	3,353	-11,078	2,347
	Equal variances not assumed			-1,285	49,336	,205	-4,366	3,397	-11,192	2,460

Ebből megállapítható, hogy a két minta a kísérlet kezdetekor homogénnek tekinthető 5 % szignifikancia szinten ( $p=0,05$  hibaszinten), vagyis nincs szignifikáns különbség a két csoport bemeneti tesztje között.



### VI.3.2.3. Károly Róbert Főiskola bemeneti tesztjei

A következő 2 táblázatban a kontroll és a kísérleti csoport bemeneti tesztjének leíró statisztikái láthatóak.

#### Kontroll csoport – Károly Róbert Főiskola

##### Descriptive Statistics

18. táblázat

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
be_szum	38	48	23	71	51,29	11,800	139,238
Valid N (listwise)	38						

#### Kísérleti csoport – Károly Róbert Főiskola

##### Descriptive Statistics

19. táblázat

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
be_szum	27	51	30	81	50,33	13,150	172,923
Valid N (listwise)	27						

Mivel mindkét minta normál eloszlású volt, valamint az F-próba is alkalmazhatóvá tette, ezért a két minta összehasonlítását 2-mintás T-próbával végeztem el.

##### Group Statistics

20. táblázat

	csoport	N	Mean	Std. Deviation	Std. Error Mean
be_szum	0	38	51,29	11,800	1,914
	1	27	50,33	13,150	2,531

21. táblázat

Independent Samples Test									
		Levene's Test for Equality of Variances		t-test for Equality of Means					
		F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference
be_szum	Equal variances assumed	,567	,454	,307	63	,760	,956	3,115	-5,268 7,181
	Equal variances not assumed			,301	52,244	,764	,956	3,173	-5,410 7,323

Ebből megállapítható, hogy a két minta a kísérlet kezdetekor homogénnek tekinthető 5 % szignifikancia szinten ( $p=0,05$  hibaszinten), vagyis nincs szignifikáns különbség a két csoport bemeneti tesztje között.

### VI.3.2.4. Mindhárom csoport bemeneti tesztjeinek összehasonlítása

Megvizsgáltam az összes kísérleti és kontroll csoportra vonatkozóan is a bemeneti tesztek eredményeit.

#### Kontroll csoport – mindhárom helyszín, összes bemeneti teszt

##### Descriptive Statistics

22. táblázat

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
be_szum	111	79	9	88	42,54	15,528	241,123
Valid N (listwise)	111						

#### Kísérleti csoport – mindhárom helyszín, összes bemeneti teszt

##### Descriptive Statistics

23. táblázat

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
be_szum	75	70	11	81	45,19	15,805	249,803
Valid N (listwise)	75						

Mivel mindkét minta normál eloszlású volt, valamint az F-próba is alkalmazhatóvá tette, ezért a két minta összehasonlítását 2-mintás T-próbával végeztem el.

##### Group Statistics

24. táblázat

	csoport	N	Mean	Std. Deviation	Std. Error Mean
be_szum	0	111	42,54	15,528	1,474
	1	75	45,19	15,805	1,825

25. táblázat

Independent Samples Test									
		Levene's Test for Equality of Variances		t-test for Equality of Means					
		F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference Lower Upper
be_szum	Equal variances assumed	,084	,772	-1,132	184	,259	-2,646	2,338	-7,258 1,966
	Equal variances not assumed			-1,128	157,059	,261	-2,646	2,346	-7,280 1,987

Ebből megállapítható, hogy a két minta a kísérlet kezdetekor homogénnek tekinthető 5 % szignifikancia szinten ( $p=0,05$  hibaszinten), vagyis nincs szignifikáns különbség a két csoport bemeneti tesztje között.

### VI.3.3. Kimeneti tesztek értékelése

#### VI.3.3.1. József Attila Szakközépiskola, Szakiskola és

#### Kollégium kimeneti tesztjei

A következő 2 táblázatban a kontroll és a kísérleti csoport kimeneti tesztjének leíró statisztikái láthatóak.

#### Kontroll csoport - József Attila Szakközépiskola, Szakiskola és Kollégium

##### Descriptive Statistics

26. táblázat

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
ki_szum	28	32	0	32	12,71	8,772	76,952
Valid N (listwise)	28						

#### Kísérleti csoport - József Attila Szakközépiskola, Szakiskola és Kollégium

##### Descriptive Statistics

27. táblázat

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
ki_szum	18	56	2	58	24,50	16,162	261,206
Valid N (listwise)	18						

Mivel a mért eredmények nem voltak egyiknél sem normál eloszlásúak, ezért a két minta összehasonlítását a Mann-Whitney nem-paraméteres (rangsorolósos) vizsgálattal hasonlítottam össze.

#### Mann-Whitney Test

##### Ranks

28. táblázat

	csoport	N	Mean Rank	Sum of Ranks
ki_szum	0	28	19,61	549,00
	1	18	29,56	532,00
	Total	46		

Test Statistics(a)

29. táblázat

	ki_szum
Mann-Whitney U	143,000
Wilcoxon W	549,000
Z	-,456
Asymp. Sig. (2-tailed)	,014

a Grouping Variable: csoport

Ebből megállapítható, hogy a két minta a kísérlet végén nem tekinthető homogénnek 5 % szignifikancia szinten ( $p=0,05$  hibaszinten), vagyis szignifikáns a különbség a két csoport kimeneti tesztje között.

### VI.3.3.2. Vak Bottyán János Katolikus Műszaki és Közgazdasági Középiskola kimeneti tesztjei

A következő 2 táblázatban a kontroll és a kísérleti csoport kimeneti tesztjének leíró statisztikái láthatóak.

#### Kontroll csoport - Vak Bottyán János Katolikus Műszaki és Közgazdasági Középiskola

Descriptive Statistics

30. táblázat

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
ki_szum	32	63	5	68	34,72	14,328	205,305
Valid N (listwise)	32						

#### Kísérleti csoport - Vak Bottyán János Katolikus Műszaki és Közgazdasági Középiskola

Descriptive Statistics

31. táblázat

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
ki_szum	26	74	8	82	44,19	20,006	400,242
Valid N (listwise)	26						

Ugyan mindkét minta eredményei normál eloszlású voltak, viszont a szórásukban akkor a különbség volt, hogy a 2-mintás T-próba nem volt elvégezhető (F-próba kizárta a használatát). Így a Welch-próbát alkalmaztam a minták összehasonlítására.

## ANOVA

32. táblázat

ki_szum	Sum of Squares	df	Mean Square	F	Sig.
Between Groups	1287,424	1	1287,424	4,404	,040
Within Groups	16370,507	56	292,330		
Total	17657,931	57			

## Robust Tests of Equality of Means

33. táblázat

ki_szum	Statistic(a)	df1	df2	Sig.
Welch	4,115	1	44,016	,049

a. Asymptotically F distributed.

Ebből megállapítható, hogy a két minta a kísérlet végén nem tekinthető homogénnek 5 % szignifikancia szinten ( $p=0,05$  hibaszinten), vagyis szignifikáns a különbség a két csoport kimeneti tesztje között.

## VI.3.3.3. Károly Róbert Főiskola kimeneti tesztjei

A következő 2 táblázatban a kontroll és a kísérleti csoport kimeneti tesztjének leíró statisztikái láthatóak.

## Kontroll csoport – Károly Róbert Főiskola

## Descriptive Statistics

34. táblázat

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
ki_szum	32	84	4	88	47,88	19,153	366,823
Valid N (listwise)	32						

## Kísérleti csoport – Károly Róbert Főiskola

## Descriptive Statistics

35. táblázat

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
ki_szum	29	67	18	85	50,72	15,257	232,778
Valid N (listwise)	29						

Mivel mindkét minta normál eloszlású volt, valamint az F-próba is alkalmazhatóvá tette, ezért a két minta összehasonlítását 2-mintás T-próbával végeztem el.

Group Statistics

36. táblázat

	csoport	N	Mean	Std. Deviation	Std. Error Mean
ki_szum	0	32	47,88	19,153	3,386
	1	29	50,72	15,257	2,833

37. táblázat

Independent Samples Test									
		Levene's Test for Equality of Variances		t-test for Equality of Means					
		F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference
									Lower Upper
ki_szum	Equal variances assumed	1,083	,302	-.638	59	,526	-2,849	4,464	-11,782 6,084
	Equal variances not assumed			-.645	58,083	,521	-2,849	4,415	-11,868 5,988

Ebből megállapítható, hogy a két minta a kísérlet végén is homogénnek tekinthető 5 % szignifikancia szinten ( $p=0,05$  hibaszinten), vagyis nincs szignifikáns különbség a két csoport kimeneti tesztje között.

### VI.3.3.4. Mindhárom csoport kimeneti tesztjeinek összehasonlítása

Megvizsgáltam az összes kísérleti és kontroll csoportra vonatkozóan is a kimeneti tesztek eredményeit.

#### Kontroll csoport – mindhárom helyszín, összes kimeneti teszt

##### Descriptive Statistics

38. táblázat

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
ki_szum	92	88	0	88	32,60	20,568	423,034
Valid N (listwise)	92						

#### Kísérleti csoport – mindhárom helyszín, összes kimeneti teszt

##### Descriptive Statistics

39. táblázat

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
ki_szum	73	83	2	85	41,93	20,003	400,120
Valid N (listwise)	73						

Mivel a mért eredmények nem voltak egyiknél sem normál eloszlásúak, ezért a két minta összehasonlítását a Mann-Whitney nem-paraméteres (rangsorolós) vizsgálattal hasonlítottam össze.

### Mann-Whitney Test

#### Ranks

40. táblázat

	csoport	N	Mean Rank	Sum of Ranks
ki_szum	0	92	73,59	6770,00
	1	73	94,86	6925,00
	Total	165		

#### Test Statistics(a)

41. táblázat

	ki_szum
Mann-Whitney U	2492,000
Wilcoxon W	6770,000
Z	-2,842
Asymp. Sig. (2-tailed)	,004

a. Grouping Variable: csoport

Ebből megállapítható, hogy a két minta a kísérlet végén nemcsak 5 % szignifikancia szinten ( $p=0,05$  hibaszinten) nem volt homogénnek tekinthető, hanem 1 % szignifikancia szinten ( $p=0,01$  hibaszinten) sem, vagyis a különbség a két csoport kimeneti tesztje között erősen szignifikáns volt.

### VI.3.4. A kísérlet értékelése

Mindhárom helyszínen a kísérleti csoportok számtanilag jobban teljesítettek a kontroll csoportoknál a kísérlet végén. A két középiskolai helyszínen azonban szignifikáns különbséget mértem a kísérleti és kontroll csoportok eredményeiben a kísérleti csoportok javára. A harmadik helyszínen (a Főiskolán) is tapasztaltam a kurzus jótékony hatását, de itt szignifikáns különbséget nem tudtam kimutatni a két csoport között.

Az összevont kísérleti és kontroll csoportok eredményeinek a vizsgálatokor kimutatható az erősen szignifikáns különbség a két csoport között.

Úgy vélem, hogy ezzel sikerült bizonyítanom a logikai programozás oktatásának a jótékony hatását a problémamegoldó képesség javításában.

Megvizsgáltam még azt is, hogy mely feladatokban sikerült szignifikánsan jobban teljesíteni a kísérleti csoport tagjainak. Először a bemeneti teszt egyes feladatainak a megoldásait hasonlítottam össze, majd pedig a kimeneti teszt feladatainak a megoldásait.

#### A kontroll csoport bemeneti tesztjének feladatonkénti leíró statisztikája:

##### Descriptive Statistics

42. táblázat

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
be_1	111	6	0	6	5,37	1,753	3,071
be_2	111	6	0	6	4,13	2,316	5,366
be_3	111	10	0	10	8,48	2,211	4,888
be_4	111	5	0	5	1,19	1,247	1,555
be_5	111	10	0	10	4,67	3,334	11,115
be_6	111	8	0	8	1,80	1,999	3,997
be_7	111	9	0	9	4,86	2,972	8,834
be_8	111	7	0	7	1,05	1,760	3,098
be_9	111	8	0	8	3,78	3,094	9,571
be_10	111	8	0	8	1,57	2,118	4,484
be_11	111	6	0	6	1,14	1,439	2,070
be_12	111	10	0	10	4,51	3,936	15,488
be_szum	111	79	9	88	42,54	15,528	241,123
Valid N (listwise)	111						

#### A kísérleti csoport bemeneti tesztjének feladatonkénti leíró statisztikája:

##### Descriptive Statistics

43. táblázat

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
be_1	75	6	0	6	4,79	2,378	5,657
be_2	75	6	0	6	4,56	2,126	4,520
be_3	75	10	0	10	8,52	2,268	5,145
be_4	75	7	0	7	1,43	1,570	2,464
be_5	75	10	0	10	5,45	3,825	14,630
be_6	75	7	0	7	2,05	2,072	4,294
be_7	75	9	0	9	5,07	3,046	9,279
be_8	75	8	0	8	1,17	2,009	4,037
be_9	75	8	0	8	4,71	3,344	11,183
be_10	75	7	0	7	1,63	1,894	3,588
be_11	75	8	0	8	1,07	1,398	1,955
be_12	75	10	0	10	4,75	4,233	17,921
be_szum	75	70	11	81	45,19	15,805	249,803
Valid N (listwise)	75						



Mivel a mért eredmények nem voltak egyiknél sem normál eloszlásúak, ezért a két minta összehasonlítását a Mann-Whitney nem-paraméteres (rangsorolósos) vizsgálatlal hasonlítottam össze.

### Mann-Whitney Test

#### Ranks

44. táblázat

	csoport	N	Mean Rank	Sum of Ranks
be_1	0	111	97,00	10767,50
	1	75	88,31	6623,50
	Total	186		
be_2	0	111	89,31	9913,50
	1	75	99,70	7477,50
	Total	186		
be_3	0	111	93,36	10363,50
	1	75	93,70	7027,50
	Total	186		
be_4	0	111	91,16	10118,50
	1	75	96,97	7272,50
	Total	186		
be_5	0	111	88,91	9869,50
	1	75	100,29	7521,50
	Total	186		
be_6	0	111	90,76	10074,50
	1	75	97,55	7316,50
	Total	186		
be_7	0	111	91,98	10209,50
	1	75	95,75	7181,50
	Total	186		
be_8	0	111	93,18	10342,50
	1	75	93,98	7048,50
	Total	186		
be_9	0	111	87,64	9728,50
	1	75	102,17	7662,50
	Total	186		
be_10	0	111	91,00	10100,50
	1	75	97,21	7290,50
	Total	186		
be_11	0	111	94,36	10474,50
	1	75	92,22	6916,50
	Total	186		
be_12	0	111	92,42	10258,50
	1	75	95,10	7132,50
	Total	186		
be_szum	0	111	89,77	9964,00
	1	75	99,03	7427,00
	Total	186		

45. táblázat

Test Statistics<sup>a</sup>

	be_1	be_2	be_3	be_4	be_5	be_6
Mann-Whitney U	3773,500	3697,500	4147,500	3902,500	3853,500	3858,500
Wilcoxon W	8623,500	9913,500	10363,500	10118,500	9869,500	10074,500
Z	-1,891	-1,439	-,044	-,752	-1,431	-,889
Asymp. Sig. (2-tailed)	,091	,150	,965	,462	,153	,385

a. Grouping Variable: csoport

	be_7	be_8	be_9	be_10	be_11	be_12	be_szum
Mann-Whitney U	3993,500	4128,500	3512,500	3884,500	4086,500	4042,500	3748,000
Wilcoxon W	10209,500	10342,500	9728,500	10100,500	8916,500	10258,500	9964,000
Z	-,473	-,122	-1,852	-,804	-,288	-,340	-1,151
Asymp. Sig. (2-tailed)	,636	,903	,064	,422	,774	,734	,260

a. Grouping Variable: csoport

Ebből megállapítható, hogy a két csoportnál a kísérlet kezdetekor a bemeneti teszt egyik feladatánál sincs szignifikáns különbség 5 % szignifikancia szinten ( $p=0,05$  hibaszinten).

#### A kontroll csoport kimeneti tesztjének feladatonkénti leíró statisztikája:

Descriptive Statistics

46. táblázat

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
ki_1	92	6	0	6	2,65	2,447	5,988
ki_2	92	6	0	6	2,50	2,434	5,923
ki_3	92	6	0	6	1,86	1,628	2,650
ki_4	92	10	0	10	5,11	4,165	17,351
ki_5	92	8	0	8	1,39	1,809	3,274
ki_6	92	7	0	7	,60	1,241	1,540
ki_7	92	10	0	10	5,15	3,774	14,240
ki_8	92	9	0	9	4,10	3,650	13,320
ki_9	92	10	0	10	,80	2,220	4,928
ki_10	92	10	0	10	4,10	4,433	19,650
ki_11	92	8	0	8	1,97	2,351	5,526
ki_12	92	8	0	8	2,37	2,116	4,477
ki_szum	92	88	0	88	32,60	20,568	423,034
Valid N (listwise)	92						

**A kísérleti csoport kimeneti tesztjének feladatonkénti leíró statisztikája:**

**Descriptive Statistics**

**47. táblázat**

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
ki_1	73	6	0	6	3,81	2,132	4,546
ki_2	73	6	0	6	3,56	2,345	5,500
ki_3	73	6	0	6	2,18	1,686	2,843
ki_4	73	10	0	10	5,77	4,293	18,431
ki_5	73	8	0	8	2,12	2,075	4,304
ki_6	73	7	0	7	,70	1,340	1,797
ki_7	73	10	0	10	6,88	3,559	12,665
ki_8	73	9	0	9	4,33	3,375	11,390
ki_9	73	10	0	10	1,22	2,755	7,590
ki_10	73	10	0	10	6,11	4,277	18,293
ki_11	73	8	0	8	2,70	2,465	6,075
ki_12	73	8	0	8	2,56	2,309	5,333
ki_szum	73	83	2	85	41,93	20,003	400,120
Valid N (listwise)	73						

Mivel a mért eredmények nem voltak egyiknél sem normál eloszlásúak, ezért a két minta összehasonlítását a Mann-Whitney nem-paraméteres (rangsorolósos) vizsgálattal hasonlítottam össze.

## Mann-Whitney Test

### Ranks

48. táblázat

	csoport	N	Mean Rank	Sum of Ranks
ki_1	0	92	73,09	6724,00
	1	73	95,49	6971,00
	Total	165		
ki_2	0	92	74,39	6844,00
	1	73	93,85	6851,00
	Total	165		
ki_3	0	92	78,66	7236,50
	1	73	88,47	6458,50
	Total	165		
ki_4	0	92	80,09	7368,00
	1	73	86,67	6327,00
	Total	165		
ki_5	0	92	74,75	6877,00
	1	73	93,40	6818,00
	Total	165		
ki_6	0	92	80,79	7432,50
	1	73	85,79	6262,50
	Total	165		
ki_7	0	92	72,89	6706,00
	1	73	95,74	6989,00
	Total	165		
ki_8	0	92	80,97	7449,00
	1	73	85,56	6246,00
	Total	165		
ki_9	0	92	80,21	7379,50
	1	73	86,51	6315,50
	Total	165		
ki_10	0	92	72,56	6675,50
	1	73	96,16	7019,50
	Total	165		
ki_11	0	92	76,76	7061,50
	1	73	90,87	6633,50
	Total	165		
ki_12	0	92	81,36	7485,50
	1	73	85,06	6209,50
	Total	165		
ki_szum	0	92	73,59	6770,00
	1	73	94,86	6925,00
	Total	165		

49. táblázat

Test Statistics <sup>a</sup>							
	ki_1	ki_2	ki_3	ki_4	ki_5	ki_6	
Mann-Whitney U	2448,000	2586,000	2958,500	3090,000	2599,000	3154,500	
Wilcoxon W	8724,000	8844,000	7236,500	7388,000	6877,000	7432,500	
Z	-3,074	-2,678	-1,344	-,908	-2,617	-,853	
Asymp. Sig. (2-tailed)	,002	,007	,179	,364	,009	,394	
a. Grouping Variable: csoport							
	ki_7	ki_8	ki_9	ki_10	ki_11	ki_12	ki_szum
Mann-Whitney U	2428,000	3171,000	3101,500	2397,500	2783,500	3207,500	2492,000
Wilcoxon W	8706,000	7449,000	7379,500	6675,500	7061,500	7485,500	6770,000
Z	-3,143	-,629	-1,165	-3,289	-1,989	-,508	-2,842
Asymp. Sig. (2-tailed)	,002	,529	,244	,001	,047	,611	,004
a. Grouping Variable: csoport							

Ebből megállapítható, hogy a két csoportnál a kísérlet végén a kimeneti teszt 1., 2., 5., 7., 10. feladatainál erősen szignifikáns különbség volt tapasztalható 1 % szignifikancia szinten ( $p=0,01$  hibaszinten). A kimeneti teszt 11. feladatánál szignifikáns a különbség 5 % szignifikancia szinten ( $p=0,05$  hibaszinten).

### VI.3.5. A kísérlet alapján levonható következtetések

Az előző rész alapján azt figyelhetjük meg, hogy a tananyag jótékony hatása mindegyik helyszínen mérhető volt. A két középiskolai helyszínen, valamint az összes helyszín összesített eredményeiben is szignifikáns különbség volt a kísérleti és kontroll csoportok kimeneti tesztjének összehasonlításakor, amely a tananyag hatásának tudható be.

Ha alaposabban megvizsgáljuk az adatokat, láthatjuk, hogy annál az iskolánál volt a legeredményesebb a hatás, ahol a gyerekek a leggyengébb matematikai előképzettséggel rendelkeztek. A másik középiskolai helyszínen (ahol ugyancsak szignifikáns különbséget mértem) jobb volt a matematikai előképzettsége a tanulónak, míg a főiskolai helyszínen (ahol nem volt szignifikáns a különbség a kísérlet végén) egyéb kereszthatás is jelen volt a kísérlet lefolytatása során (matematikai felkészítő).

Ebből következően úgy látom, hogy a logikai programozást, és így a Prolog nyelvet, érdemes lenne bevezetni a középiskolai oktatásba. A Prolog nyelv problémamegoldó képességet fejlesztő eszközzé válhatna a középiskolában. Olyan szerepet tölthetne be, mint a Logo az általános iskolákban, tehát szemléletformáló eszköz lehetne.

Ezen gondolat alapján 2011 novemberében 120 középiskolának küldtem ki e-mail-ben a H függelékben látható kérdőívet. Az érdekelt elsősorban, hogy tanítják-e valahol a Prolog nyelvet illetve a Mesterséges intelligenciát. Emellett kíváncsi voltam arra is, hogy mennyire elterjedt a középiskolákban az e-learning, tanítják-e az SQL nyelvet, s ha tanítanak programozást, akkor milyen programnyelveket oktatnak.

21 középiskolából érkezett válasz a megkeresésre, amelyet a kérdezési módhoz képest jó aránynak tartok (a szakirodalom kb. 10%-os kitöltést említ a kérdőívek elektronikus módon történő kitöltésére).

A válaszokból egyértelműen kiderült, hogy a kérdőívet kitöltött iskolákban Prolog nyelvet sehol sem tanítanak, mesterséges intelligencia kurzus sincs sehol. E-learning rendszer már található néhány iskolában. A programozási nyelveknél a Pascal vagy a C/C++/C# nyelveket oktatják döntően. Az SQL-t, mivel az emelt szintű informatika érettségi része, sok iskolában oktatják.

**Ezek alapján az a javaslatom, hogy a Prolog nyelvet be kellene illeszteni a középiskolai informatika tananyagába, mégpedig az SQL tananyagrészek után.**

## VII. Összefoglalás

Dolgozatom témája a mesterséges intelligencia elemekkel támogatott programozás oktatásának módszertana. Fő célom a tanulók problémamegoldó képességének fejlesztése, melyet a logikai programozás oktatásával megvalósíthatónak gondolok.

A problémamegoldó gondolkodás fejlesztése nagyon fontos feladat minden oktatási szinten. A problémamegoldás fejlesztésére nagyon hasznos módszernek bizonyul a logikai programozás oktatása.

Munkám során olyan intelligens tananyagot fejlesztettem ki, amelynek segítségével a tanulók természettudományos érdeklődését javítani lehet. A logikai programozás oktatására programozott tananyag-egységeket dolgoztam ki, amelyeket e-learning segítségével a modern kor követelményeinek megfelelően lehet közzétenni. Kidolgoztam a Mesterséges intelligencia, valamint a Logikai programozás e-learning-es kurzusok tananyagát.

2011 őszén összetett kétcsoportos pedagógiai kísérlet került megvalósításra három gyöngyösi oktatási intézményben (Vak Bottyán János Katolikus Műszaki és Közgazdasági Középiskola, József Attila Szakközépiskola, Szakiskola és Kollégium, Károly Róbert Főiskola). Mindegyik intézményben volt egy kísérleti csoport, valamint egy kontroll csoport, 20-30 fős létszámokkal.

A kísérleti csoportok heti két órában, közel 3 hónapon keresztül logikai programozást tanultak. A kurzusok e-learning rendszerrel kerültek támogatásra, a **<http://patronus.karolyrobert.hu>** portálon elérhetővé tett Logikai programozás kurzussal.

A kurzus elején és a kurzus végén problémamegoldó feladatsorok (bemeneti és kimeneti teszt) kerültek kitöltésre mind a három helyszínen, mindegyik csoport által. A bemeneti és a kimeneti tesztek eredményeit statisztikailag elemeztem SPSS programcsomag segítségével. Mindhárom helyszínen a kísérleti csoportok jobban teljesítettek a kontroll csoportoknál. A két középiskolai helyszínen szignifikáns különbséget mértem az eredményekben.

A kísérlet segítségével sikerült bizonyítani a logikai programozás jótékony hatását a problémamegoldó képesség javításában, így bevezetését indokoltnak tartom a középiskolai oktatásban.

## VIII. Summary

The topic of my dissertation is the methodology of teaching of programming by the utilisation of artificial intelligence. My main aim is to develop the problem solving skills of students by teaching logical programming.

The development of problem solving thinking is crucial at all levels of education. Teaching logical programming languages prove to be suitable for developing problem solving skills.

During my work I developed an intelligent curriculum which can help to improve students' interest in natural sciences. I worked out programmed teaching units to teach logical programming, which can be disclosed by the help of e-learning according to the modern requirements. I worked out the e-learning curriculums of Artificial Intelligence and Logical programming.

In the autumn of 2011 I performed a combined, two-group pedagogical experiment in 3 educational institutions in Gyöngyös (Vak Bottyán János Catholic Secondary School of Technology and Economics, József Attila Secondary Vocational School, Károly Róbert College). In each educational institution there was an experimental and a control group with 20-30 students.

The experimental groups learnt logical programming twice a week through three months. The courses were supported by an e-learning system and the course of Logical Programming was available at the portal

**<http://patronus.karolyrobert.hu>.**

At the beginning and at the end of the courses all groups wrote a problem solving test (input and output tests) at the three locations. I analysed the results of the input and output tests with the help of SPSS programme. The experiment groups performed better than the control groups at all locations. I measured significant difference in the results at two secondary institutions.

By the help of this experiment I managed to prove the beneficial effect of logical programming in developing problem-solving skills, so introducing it into secondary education is justified.



## IX. Irodalomjegyzék

- [1] Futó I. (1999): Mesterséges intelligencia, Aula Kiadó
- [2] Fekete I., Gregorics T., Nagy S. (1999): Bevezetés a mesterséges intelligenciába GDF Főiskolai jegyzet, LSI Oktatóközpont, Budapest
- [3] S. J. Russell – P. Norvig (2000): Mesterséges intelligencia modern megközelítésben, Panem Könyvkiadó
- [4] J. Haugenland (1985): The Artificial Intelligence: The Very Idea, MIT Press, Cambridge/Bradford MA
- [5] R.E.Bellmann (1978): An Introduction to Artificial Intelligence: Can Computers Think? Boyd and Fraser Publishing Co., San Francisco
- [6] E. Charniak, D. McDermott (1985): Introduction to Artificial Intelligence, Reading, MA, Addison-Wesley
- [7] P.H. Winston (1992): Artificial Intelligence, Addison-Wesley MA
- [8] R. Kurzweil (1990): The Age of Intelligent Machines, MIT Press, Cambridge, Massachusetts
- [9] E. Rich, K. Knight (1991): Artificial Intelligence, Addison Wesley Longman Inc.
- [10] R. J. Schalkoff (1990): Artificial Intelligence: An Engineering Approach, McGraw-Hill, New York
- [11] G.F. Luger, W.A. Stubblefield (1993): Artificial Intelligence, Addison Wesley Longman Inc.
- [12] Borgulya I. (1995): Szakértői rendszerek technikák és alkalmazások. ComputerBooks Kiadói Kft., Budapest
- [13] E.G. Mallach (1994): Understanding decision support system and expert systems. Irwin, Inc. New York
- [14] P. Harmon, B. Sawyer (1990): Creating expert systems for business and industry. Wiley&Sons, Inc.
- [15] Sántáné-Tóth E. (2000): Tudásalapú technológia, szakértő rendszerek, Dunaújvárosi Főiskola Kiadói Hivatala

- [16] Smith, B. C. (1982): Reflection and Semantics in a Procedural Language, Ph.D. Thesis and Tech. Report MIT/LCS/TR-272, MIT, Cambridge, MA
- [17] Pántya R. (2006): A mesterséges intelligencia szerepe a szakértői rendszerek kialakításában, működtetésében. X. Nemzetközi Agrárökonómiai Tudományos Napok. Károly Róbert Főiskola, Gyöngyös. ISBN 963 229 623 0 (CD-lemezen)
- [18] Fazekasné Kis M. (1996): Az egészségügyi szakértői rendszerek a DOTE-n folyó informatikai oktatás szemszögéből. Informatika a Felsőoktatásban'96-Networkshop'96. Debrecen, 1996.augusztus 27-30.
- [19] Szlávi P., Zsakó L. (1997): Programozási nyelvek: Alapfogalmak. műlógia 22, ELTE TTK Informatikai Tanszékcsoport, Budapest
- [20] Szeredi P, Benkő T. (2004): Deklaratív programozás, Oktatási segédlet, Bevezetés a logikai programozásba, Budapesti Műszaki és Gazdaságtudományi Egyetem, Villamosmérnöki és Informatikai Kar
- [21] <http://merd.sourceforge.net/pixel/language-study/diagram.html>
- [22] Makány Gy. (1995): Programozási nyelvek: Prologika. műlógia 7, ELTE Általános Számítástudományi Tanszék, Budapest
- [23] Márkus Zs. (1988): Prologban programozni könnyű. Novotrade, Budapest.
- [24] <http://www.felvi.hu>
- [25] STADAT 4.7.14. Háztartások info-kommunikációs eszközellátottsága és egyéni használat jellemzői (2005 – )  
[http://portal.ksh.hu/pls/ksh/docs/hun/xstadat/xstadat\\_eves/i\\_oni006.html](http://portal.ksh.hu/pls/ksh/docs/hun/xstadat/xstadat_eves/i_oni006.html)
- [26] STADAT 4.7.8. Az internet-előfizetések száma hozzáférési szolgáltatások szerint (2003 – )  
[http://portal.ksh.hu/pls/ksh/docs/hun/xstadat/xstadat\\_eves/i\\_oni001.html](http://portal.ksh.hu/pls/ksh/docs/hun/xstadat/xstadat_eves/i_oni001.html)
- [27] Felnőttoktatási Kislexikon, Budapest, 1987, Kossuth Könyvkiadó, 264. p
- [28] Kokovay Á. (2007): Multimédiás lehetőségek a testnevelés oktatásmódszertanában. Doktori értekezés, Semmelweis Egyetem, Sporttudományi Doktori Iskola

- [29] A. Vincent (1990): Molekuláris szimmetria és csoportelmélet. Programozott bevezetés a kémiai alkalmazásokba, Tankönyvkiadó, Budapest.
- [30] L.N.Landa (1966): Az algoritmusok és a programozott oktatás, Tankönyvkiadó, Budapest
- [31] Pedagógiai lexikon III. Budapest, 1997, Keraban Kiadó, 213. p.
- [32] Karlovitz J. (2001): Tankönyvtípusok, tankönyvmodellek, Új Pedagógiai Szemle 2001/01
- [33] Négyesi I., Kelemenné Bistyák K. (2000): NYITOTT EGYETEM?! ÚJ TÁVLATOK A ZMNE KÉPZÉSI RENDSZERÉBEN?! Nemzetvédelmi egyetemi FÓRUM 2000/10.
- [34] Ratkó I. (1999): Programozott oktatás lehetőségei, Informatika a felsőoktatásban konferencia, Debrecen 1999.08.27-29.
- [35] Komenczi B (2008): Egy e-learning didaktika oktatásméleti alapjai, Networkshop 2008 konferencia Dunaújvárosi Főiskola, Dunaújváros 2008. március 17 - 19.
- [36] R.C. Clark and R.E. Mayer (2003): ELearning and the science of instruction; Proven guidelines for consumers and designers of multimedia learning. San Francisco: Pfeiffer.
- [37] Komenczi B (2004): Didaktika elektromagna? Az e-learning virtuális valóságai. Új Pedagógiai Szemle, 2004. november. Oktatókutatató és -fejlesztő Intézet, Budapest
- [38] M.J. Rosenberg (2001): E-learning. Strategies for Delivering Knowledge into the Digital Age, McGraw-Hill, USA
- [39] Vágvolgyi Cs. - Papp Gy. (2007), Integrációs lehetőségek a Moodle keretrendszerben, Networkshop 2007, Eger, 2007.04.11-13.
- [40] Komló Cs. (2008), Elektronikus tananyagfejlesztés az Eszterházy Károly Főiskolán, INFODIDACT 2008, Szombathely, 2008.04.11-12.
- [41] Pántya R. – Mucsics F. L. (2008): Online tanítás, tanulás módszertana. „Zöld út a felnőttképzéshez”, Önkormányzatok és felnőttképzési intézmények partnersége a felnőttképzési szolgáltatások területén, Gyöngyös, 2008. október 28.

- [42] Forgó S., Hauser Z. és Kis-Tóth L. (2005), A blended learning elméleti és gyakorlati kérdései, Networkshop 2005, Szeged, 2005.03.30.-04.01.
- [43] Bessenyei I., Tóth Zs. (2008): A konstruktivista oktatás környezete és a Moodle, Nyugat-Magyarországi Egyetem, Sopron
- [44] Bakó M. (2000): Számítógéppel segített oktatás a gyakorlatban? Iskolakultúra 2000/9.
- [45] Bodonyi F. (1987): Kémiai összefoglaló, Műszaki Könyvkiadó, Budapest , 59-102. pp.
- [46] Shalfield, R., Spenser, C., Steel, B., and Westwood, A. (2007), WIN-PROLOG User Guide, Logic Programming Associates Ltd., London, 2007.
- [47] Sterling, L., Shapiro, L.: The Art of Prolog, Advanced Programming Techniques, The MIT Press Massachusetts Institute of Technology Cambridge, Massachusetts, 1990.
- [48] Strong, J. A. (1986): The Periodic Table and Electron Configurations, Journal of Chemical Education, Volume 63 Number 10, 834-836 pp.
- [49] Brücher E. (1992): Általános kémia (Anyagszerkezet), oktatási segédanyag, KLTE Szervetlen és Analitikai Kémiai Tanszéke, Debrecen, 108 p.
- [50] Tökés B., Donáth-Nagy G. (2002): Kémiai előadások és laboratóriumi gyakorlatok, Scientia Kiadó, Kolozsvár
- [51] Birk, J. P. (1992): The Computer as Student – An Application of Artificial Intelligence, Journal of Chemical Education, Volume 69 Number 4, 294-295 pp.
- [52] Négyjegyű függvénytáblázatok. Matematikai, fizikai, kémiai összefüggések. Tankönyvkiadó, Budapest, 1983.
- [53] Siposné Kedves É., Horváth B., Péntek Lászlóné: KÉMIA 9. Általános kémiai ismeretek, Mozaik Kiadó, Szeged, 2003.
- [54] <http://mentor.karolyrobert.hu>

- [55] Pántya R. - Muicsics F. L. (2008): Problémamegoldó gondolkodás fejlesztése logikai programozás segítségével. Felsőfokú alapképzésben matematikát, fizikát és informatikát oktatók XXXII. konferenciája, Kecskemét, 2008. augusztus 25-27.
- [56] Máday Á. (2000): Prolog feladatgyűjtemény (szakdolgozat), ELTE
- [57] Zs. Papp-Varga, P. Szilágyi and L. Zsákó (2008): ICT teaching methods – Programming languages. *Annales Mathematicae et Informaticae* Volume 35, pp. 163-172.
- [58] <http://www.dynamicdrive.com/dynamicindex12/towerhanoi.htm>
- [59] <http://egyszervolt.hu/jatek/jatek-hanoi-tornyai.html>
- [60] Pántya R. (2008): Programozott tananyag Prolog alkalmazások készítéséhez. I. Informatika Szakmódszertani Konferencia - INFODIDACT 2008, Szombathely 2008. április 11-12.
- [61] R. Pántya and L. Zsákó (2008): Computer-Based Intelligent Educational Program for Teaching Chemistry. *Acta Cybernetica*, Volume 18 Number 4, 595-613 p.
- [62] <http://patronus.karolyrobert.hu>
- [63] Róka S.: Hány éves a kapitány? Typotex kiadó, Budapest, 2006
- [64] Róka S.: Szakköri feladatok matematikából 5-6.osztály, Tóth Könyvkereskedés és Kiadó Kft., Debrecen, 2002
- [65] Róka S.: Szakköri feladatok matematikából 7-8.osztály, Tóth Könyvkereskedés és Kiadó Kft., Debrecen, 2002
- [66] Zrínyi Ilona Matematikaverseny feladatai 1992-2000. 4.osztály, Mategye Alapítvány, Kecskemét, 2007.
- [67] Zrínyi Ilona Matematikaverseny feladatai 1992-2000. 5.osztály, Mategye Alapítvány, Kecskemét, 2007.
- [68] Zrínyi Ilona Matematikaverseny feladatai 1992-2000. 6.osztály, Mategye Alapítvány, Kecskemét, 2007.
- [69] R. Smullyan: Mi a címe ennek a könyvnek? Typotex kiadó, Budapest, 1998
- [70] R. Smullyan: A hölgy vagy a tigris? és egyéb logikai feladatok, Typotex kiadó, Budapest, 1997

- [71] <http://oecd-pisa.hu/feladatok/>
- [72] Falus I. – Ollé J.: Statisztikai módszerek pedagógusok számára, Okker Kiadó, 2000.
- [73] Sajtos L. – Mitev A.: SPSS kutatási és adatelemzési kézikönyv, Alinea Kiadó, Budapest, 2007.
- [74] <http://www.swi-prolog.org/>
- [75] <http://tablajatekos.hu/uj2001/2003/flash/nim.html>
- [76] <http://freeweb.siol.net/danej/riverIQGame.swf>

## **X. A szerző disszertációhoz kapcsolódó publikációi**

1. R. Pántya and L. Zsakó (2008): Computer-Based Intelligent Educational Program for Teaching Chemistry. Acta Cybernetica, Volume 18 Number 4, 595-613 p.
2. R. Pántya and L. Mucsics F. (2010): Increasing the popularity and efficiency of distance education by old-new methods. Teaching Mathematics and Computer Science, Volume 8 Issue 2, 211-228 p.
3. Pántya R. (2006): A mesterséges intelligencia szerepe a szakértői rendszerek kialakításában, működtetésében. X. Nemzetközi Agrárökonómiai Tudományos Napok. Károly Róbert Főiskola, Gyöngyös. ISBN 963 229 623 0 (CD-lemezen)
4. Pántya R. (2007): Intelligens feladatmegoldási lehetőségek PROLOG segítségével. Felsőoktatási Matematika-, Fizika- és Számítástechnika Oktatók XXXI. konferenciája, Dunaújvárosi Főiskola, Dunaújváros, 2007. augusztus 23-25.
5. Pántya R. (2007): Intelligens segédanyag a kémia oktatásához (poszter). Intelligens Rendszerek 2007 – Fiatal Kutatók 2. Szimpóziuma (IRFIX'07), Budapesti Műszaki Főiskola Neumann János Informatikai Kara, Budapest 2007. november 23.
6. Pántya R. – Mucsics F. L. (2008): Online tanítás, tanulás módszertana. „Zöld út a felnőttképzéshez”, Önkormányzatok és felnőttképzési intézmények partnersége a felnőttképzési szolgáltatások területén (előadás), Gyöngyös, 2008. október 28.
7. Pántya R. – Mucsics F. L. (2008): Logikai programozás oktatása e-learning rendszer támogatásával (poszter). Intelligens Rendszerek 2008 – Fiatal Kutatók 3. Szimpóziuma (IRFIX'08), Budapesti Műszaki Főiskola Neumann János Informatikai Kara, Budapest 2008. november 28.
8. Pántya R. (2008): Programozott tananyag Prolog alkalmazások készítéséhez. I. Informatika Szakmódszertani Konferencia - INFODIDACT 2008, Szombathely 2008. április 11-12.

9. Pántya R. - Mucsics F. L. (2008): Problémamegoldó gondolkodás fejlesztése logikai programozás segítségével. Felsőfokú alapképzésben matematikát, fizikát és informatikát oktatók XXXII. konferenciája, Kecskemét, 2008. augusztus 25-27.
10. Pántya R. - Mucsics F. L. (2009): A távoktatási képzési forma hatékonyságának és népszerűségének növelése régi-új módszerekkel . II. Informatika Szakmódszertani Konferencia - INFODIDACT 2009, Szombathely 2009. április 23-24.
11. Pántya R. - Mucsics F. L. (2009): Programozott oktatás e-learning környezetben. Matematikát, fizikát és informatikát oktatók XXXIII. országos és nemzetközi konferenciája, Budapest, 2009. augusztus 24-26.
12. Mucsics F. L. - Pántya R. (2009): First attempts of launching blended learning courses at Károly Róbert College, Gyöngyös. Thüringisch-Ungarisches Symposium 2009, ISBN 978-3-932886-21-8, Fachhochschule Jena, pp 144-157
13. Pántya R., Mucsics F. L. (2009): How can MOODLE e-learning environment support teaching/learning activities? Dulama, Maria Eliza, Bucila, F., Ilovan, Oana-Ramona (eds.) (2009), Tendinte actuale in predarea si invatarea geografiei / Contemporary Trends in Teaching and Learning Geography, Romania ISBN 978-973-610-924-9 pp. 421-432
14. Mucsics F. L. - Pántya R. (2010): The early period of e-learning history at Károly Róbert College. XII. Nemzetközi Tudományos Napok. Károly Róbert Főiskola, Gyöngyös, 2010. március 25-26.
15. Pántya R., Mucsics F. L., Tóth. Z. (2010): Blended learning kurzusok a Károly Róbert Főiskola Gazdaságmatematika és informatika Tanszékének gondozásában. Matematikát, fizikát és informatikát oktatók (MAFIOK) XXXIV. konferenciája, Békéscsaba, 2010. augusztus 24-26.
16. Mucsics F. L. – Pántya R. (2011): E-learning kézikönyv, Károly Róbert Főiskola, Gyöngyös. TÁMOP-4.1.1/A-10/1/KONV-2010-0010 azonosító számú, „Az Észak-magyarországi régió hiányszakmáira épülő gyakorlati képzőhelyek kialakítása, valamint intézményi szolgáltatásfejlesztés a Károly Róbert Főiskolán.” projekt keretében.
17. Mucsics F. L. - Pántya R. (2011): Handbook of E-learning Applications – prerequisites, tools, results. Thüringisch-Ungarisches Symposium 2011, ISBN 978-3-932-886-28-7, Fachhochschule Jena, pp 173-177



## XI. Függelék

### A függelék – Az intelligens tananyag forráskódja

```
intelligens :- hangol1,
_S1 = [ws_caption,ws_sysmenu,dlg_ownedbyprolog],
_S2 = [ws_child,ws_visible,ss_left],
_S3 =
[ws_child,ws_tabstop,ws_visible,bs_pushbutton,bs_text,bs_center,bs_vcenter],
_S4 = [ws_child,ws_border,ws_visible],
wdcreate( intelligens, `Intelligens segédanyag a kémia oktatásához`, 263, 87,
586, 195, _S1 ),
wcreate( (intelligens,11001), static, `Készítette:`, 10, 140, 60, 20, _S2 ),
wcreate( (intelligens,11002), static, `Pántya Róbert, Dr.Zsakó László`, 70, 140,
190, 20, _S2 ),
wcreate( (intelligens,1000), button, `Kvantumszámok`, 340, 20, 110, 30, _S3 ),
wcreate( (intelligens,1001), button, `Elektronszerkezet`, 340, 70, 110, 30, _S3 ),
wcreate( (intelligens,1002), button, `Oxidációs számok`, 460, 20, 110, 30, _S3 ),
wcreate( (intelligens,1003), button, `Kilépés`, 460, 120, 110, 30, _S3 ),
wcreate( (intelligens,10001), grafix, `Grafix2`, 10, 20, 316, 104, _S4 ),
wcreate( (intelligens,1004), button, `Periódusos rendszer`, 340, 120, 110, 30, _S3 ),
wcreate( (intelligens,1005), button, `Elektronegativitás`, 460, 70, 110, 30, _S3 ).
hangol1:-
    absolute_file_name( examples('kémia50.bmp'), File1 ),
    gfx_bitmap_load( rajz1, File1 ), !.
kérdés1:-intelligens,
    window_handler( intelligens, intelligens_handler ),
    call_dialog(intelligens, _).
intelligens_handler( intelligens, msg_close, _, done ).
intelligens_handler( (intelligens,1003), msg_button, _, done ).
intelligens_handler( (intelligens,1000), msg_button, _, _):- kérdés2_1.
intelligens_handler( (intelligens,1001), msg_button, _, _):- kérdés6.
intelligens_handler( (intelligens,1002), msg_button, _, _):- kérdés8.
intelligens_handler( (intelligens,1004), msg_button, _, _):- kérdés3.
```

```

intelligens_handler( (intelligens,1005), msg_button, _, _):- kérdés9.
intelligens_handler( (intelligens,10001), msg_paint, grafix, _ ):-
gfx_paint( (intelligens,10001) ), gfx( bitmap(0,0,2000,2000,0,0,rajz1) ),
gfx_end( (intelligens,10001) ).
kvantum :-
    _S1 = [ws_caption,ws_sysmenu,dlg_ownedbyprolog],
    _S2 =
[ws_child,ws_border,ws_disabled,ws_tabstop,ws_visible,es_left,es_multiline,es_a
utohscroll,es_autovscroll],
    _S3 =
[ws_child,ws_tabstop,ws_visible,bs_pushbutton,bs_text,bs_center,bs_vcenter],
    wdcreate( kvantum, 'Kvantumszámok', 269, 83, 616, 495, _S1 ),
    wccreate( (kvantum,9000), rich, '', 10, 10, 590, 410, _S2 ),
    wccreate( (kvantum,1000), button, 'Héjak', 10, 430, 100, 30, _S3 ),
    wccreate( (kvantum,1001), button, 'Héjak és alhéjak', 130, 430, 140, 30, _S3 ),
    wccreate( (kvantum,1002), button, 'Pályák és elektronok száma', 290, 430, 180,
30, _S3 ),
    wccreate( (kvantum,1003), button, 'Vissza', 490, 430, 110, 30, _S3 ).
kérdés2_1:- kvantum, absolute_file_name( examples('szoveg2.rtf'), File2 ),
    wrchlod((kvantum,9000),1,File2), window_handler( kvantum,
kvantum_handler ),
    call_dialog(kvantum, _).
kvantum_handler( kvantum, msg_close, _, done ).
kvantum_handler( (kvantum,1003), msg_button, _, done ).
kvantum_handler( (kvantum,1000), msg_button, _, _):- kérdés4.
kvantum_handler( (kvantum,1001), msg_button, _, _):- kérdés5.
kvantum_handler( (kvantum,1002), msg_button, _, _):- kérdés2.
pályák :-
    _S1 = [ws_caption,ws_sysmenu,dlg_ownedbyprolog],
    _S2 = [ws_child,ws_visible,ss_left],
    _S3 = [ws_child,ws_tabstop,ws_visible,cbs_dropdownlist],
    _S4 = [ws_child,ws_border,ws_tabstop,ws_visible,ws_vscroll],
    _S5 =
[ws_child,ws_tabstop,ws_visible,bs_pushbutton,bs_text,bs_center,bs_vcenter],

```

```
_S6 =
[ws_child,ws_border,ws_disabled,ws_tabstop,ws_visible,es_left,es_multiline,es_a
utohscroll,es_autovscroll],
```

```
wdcreate( pályák, 'Pályák és elektronok', 259, 87, 576, 485, _S1 ),
wcreate( (pályák,11000), static, 'Kérem a héj kiválsztását!', 260, 20, 130,
20, _S2 ),
```

```
wcreate( (pályák,5000), combobox, 'Combo1', 390, 20, 110, 150, _S3 ),
```

```
wcreate( (pályák,11001), static, 'fő (n)', 40, 20, 30, 20, _S2 ),
```

```
wcreate( (pályák,4000), listbox, 'List1', 40, 50, 20, 140, _S4 ),
```

```
wcreate( (pályák,11002), static, 'mellék (l)', 80, 20, 50, 20, _S2 ),
```

```
wcreate( (pályák,4001), listbox, 'List2', 90, 50, 20, 140, _S4 ),
```

```
wcreate( (pályák,11003), static, 'pályák', 130, 20, 40, 20, _S2 ),
```

```
wcreate( (pályák,4002), listbox, 'List3', 140, 50, 20, 140, _S4 ),
```

```
wcreate( (pályák,11004), static, 'elektronok', 170, 20, 60, 20, _S2 ),
```

```
wcreate( (pályák,4003), listbox, 'List4', 190, 50, 20, 140, _S4 ),
```

```
wcreate( (pályák,1000), button, 'megjelenít', 400, 90, 100, 30, _S5 ),
```

```
wcreate( (pályák,1001), button, 'vissza', 400, 140, 100, 30, _S5 ),
```

```
wcreate( (pályák,9000), rich, '', 10, 190, 550, 260, _S6 ).
```

kérdés2:- pályák,

```
absolute_file_name( examples('pályák+elektronok.rtf'), File8 ),
```

```
wrchlod((pályák,9000),1,File8),
```

```
wcmbadd((pályák,5000),-1,'1',0),
```

```
wcmbadd((pályák,5000),-1,'2',0),
```

```
wcmbadd((pályák,5000),-1,'3',0),
```

```
wcmbadd((pályák,5000),-1,'4',0),
```

```
wcmbadd((pályák,5000),-1,'5',0),
```

```
wcmbadd((pályák,5000),-1,'6',0),
```

```
wcmbadd((pályák,5000),-1,'7',0),
```

```
wcmbset((pályák,5000),0,0),
```

```
window_handler( pályák, pályák_handler ),
```

```
call_dialog(pályák, _).
```

```
pályák_handler( pályák, msg_close, _, done ).
```

```
pályák_handler( (pályák,1001), msg_button, _, done ).
```

```
pályák_handler( (pályák,1000), msg_button, _, _):-
```

```

        wlbxget((pályák,5000),Pos), number_string(N2,Pos), L2 is 0,
        elektronok2(N2, L2),új2.
új2:-   újra2,
        window_handler( újra2, újra2_handler ),
        call_dialog(újra2, _).
újra2_handler( (újra2,1000), msg_button, _, _):- kérdés2.
újra2_handler( (újra2,1001), msg_button, _, _):- wclose(újra2), wclose(pályák),
kérdés2_1.

új:-   újra, window_handler( újra, újra_handler ), call_dialog(újra, _).
újra_handler( (újra,1000), msg_button, _, _):- kérdés5.
újra_handler( (újra,1001), msg_button, _, _):- wclose(újra), wclose(alhÉjak),
kérdés2_1.
újra :-
    _S1 = [ws_caption,ws_sysmenu,dlg_ownedbyprolog],
    _S2 = [ws_child,ws_visible,ss_center],
    _S3 =
[ws_child,ws_tabstop,ws_visible,bs_pushbutton,bs_text,bs_center,bs_vcenter],
    wdcreate( újra, 'Visszalép', 291, 112, 296, 125, _S1 ),
    wccreate( (újra,11000), static, 'Folytatja más adatokkal?', 100, 10, 130, 20,
    _S2 ),
    wccreate( (újra,1000), button, 'Igen', 20, 50, 110, 30, _S3 ),
    wccreate( (újra,1001), button, 'Nem', 160, 50, 110, 30, _S3 ).
újra2 :-
    _S1 = [ws_caption,ws_sysmenu,dlg_ownedbyprolog],
    _S2 = [ws_child,ws_visible,ss_center],
    _S3 =
[ws_child,ws_tabstop,ws_visible,bs_pushbutton,bs_text,bs_center,bs_vcenter],
    wdcreate( újra2, 'Visszalép', 499, 277, 316, 115, _S1 ),
    wccreate( (újra2,11000), static, 'Folytatja más adatokkal?', 90, 10, 130, 20,
    _S2 ),
    wccreate( (újra2,1000), button, 'Igen', 30, 40, 110, 30, _S3 ),
    wccreate( (újra2,1001), button, 'Nem', 170, 40, 110, 30, _S3 ).
elektronok2(N, L):- L=N, show_dialog(pályák).
elektronok2(N, L):- L<N, number_string(N, N3), wlstadd((pályák,4000),-1,
N3,0),

```

```

        number_string(L, L3), wlstadd((pályák,4001),-1,L3,0),
        P is 2*L+1,    number_string(P, P2), wlstadd((pályák,4002),-
1,P2,0),
        E is 2*P,      number_string(E, E2), wlstadd((pályák,4003),-
1,E2,0),

```

L4 is L+1, elektronok2(N,L4).

wlbgxget(Window, String):- wlbgxget1(Window, 0, String).

wlbgxget1(Window, Posn, Selected):-

```

        wlbgxget(Window, Posn, String),
        wlbgxsel(Window, Posn, Sel),
        (Sel=1
-> Selected = String;
        Sel=0
-> NextPosn is Posn+1,
        wlbgxget1(Window, NextPosn, Selected)).

```

image\_periodic :- hangol2,

```

    _S1 = [ws_caption,ws_sysmenu,dlg_o],
    _S6 =
[ws_child,ws_border,ws_disabled,ws_tabstop,ws_visible,es_left,es_multiline,es_a
utohscroll,es_autovscroll],
    _S7 = [ws_child,ws_border,ws_visible],
    wdcreate( alhéjak,    `Héjak és alhéjak`,          223, 52, 666, 625, _S1 ),
    wccreate( (alhéjak,11000), static,  `Kérem a héj kiválsztását!`, 520, 420, 130,
20, _S2 ),
    wccreate( (alhéjak,5000), combobox, `Combo1`,      540, 440, 110, 150, _S3 ),
    wccreate( (alhéjak,11001), static,  `fő (n)`,       420, 430, 30, 20, _S2 ),
    wccreate( (alhéjak,4000), listbox,  `List1`,        420, 460, 20, 140, _S4 ),
    wccreate( (alhéjak,11002), static,  `mellék (l)`,    460, 430, 50, 20, _S2 ),
    wccreate( (alhéjak,4001), listbox,  `List2`,        480, 460, 20, 140, _S4 ),
    wccreate( (alhéjak,1000), button,   `megjelenít`,   550, 520, 100, 30, _S5 ),
    wccreate( (alhéjak,1001), button,   `vissza`,       550, 560, 100, 30, _S5 ),
    wccreate( (alhéjak,9000), rich,     ``,             10, 410, 360, 180, _S6 ),
    wccreate( (alhéjak,10000), grafix,  `Grafix1`,      10, 10, 640, 390, _S7 ).

```

hangol4:-

```

absolute_file_name( examples('periodic_table_alhéjak.bmp'), File6 ),

```

```

gfx_bitmap_load( rajz4, File6 ), !.
kérdés5:- alhéjak,
            absolute_file_name( examples('héjak+alhéjak.rtf'), File7 ),
            wrchlod((alhéjak,9000),1,File7),
            wcmbadd((alhéjak,5000),-1,'1',0),
            wcmbadd((alhéjak,5000),-1,'2',0),
            wcmbadd((alhéjak,5000),-1,'3',0),
            wcmbadd((alhéjak,5000),-1,'4',0),
            wcmbadd((alhéjak,5000),-1,'5',0),
            wcmbadd((alhéjak,5000),-1,'6',0),
            wcmbadd((alhéjak,5000),-1,'7',0),
            wcmbasel((alhéjak,5000),0,0),
            window_handler( alhéjak, alhéjak_handler ),
            call_dialog(alhéjak, _).
alhéjak_handler( alhéjak, msg_close, _, done ).
alhéjak_handler( (alhéjak,1001), msg_button, _, done ).
alhéjak_handler( (alhéjak,1000), msg_button, _, _):-
            wlbxget((alhéjak,5000),Pos), number_string(N2,Pos), L2 is 0,
            elektronok(N2, L2),új.
alhéjak_handler( (alhéjak,10000), msg_paint, grafix, _ ):-
            gfx_paint( (alhéjak,10000) ),
            gfx( bitmap(0,0,2000,2000,0,0,rajz4) ),
            gfx_end( (alhéjak,10000) ).
elektronok(N, L):- L=N, show_dialog(alhéjak).
elektronok(N, L):- L<N, number_string(N, N3), wlstadd((alhéjak,4000),-1, N3,0),
            number_string(L, L3), wlstadd((alhéjak,4001),-1,L3,0),
            L4 is L+1, elektronok(N,L4).
e_szerkezet :-
            _S1 = [ws_caption,ws_sysmenu,dlg_ownedbyprolog],
            _S2 =
            [ws_child,ws_tabstop,ws_visible,bs_pushbutton,bs_text,bs_center,bs_vcenter],
            _S3 =
            [ws_child,ws_border,ws_tabstop,ws_visible,es_left,es_multiline,es_autohscroll,es_auto
            vscroll],
            _S4 = [ws_child,ws_visible,ss_left],

```

```

    _S5 = [ws_child,ws_border,ws_tabstop,ws_visible,ws_vscroll],
    _S6 =
[ws_child,ws_border,ws_disabled,ws_tabstop,ws_visible,es_left,es_multiline,es_a
utohscroll,es_autovscroll],

    wdcreate( e_szerkezet, 'Elemek elektronszerkezete', 286, 102, 586, 565, _S1 ),
    wccreate( (e_szerkezet,1000), button, 'Számol', 400, 320, 110, 30, _S2 ),
    wccreate( (e_szerkezet,1001), button, 'Vissza', 400, 420, 110, 30, _S2 ),
    wccreate( (e_szerkezet,8000), edit, '', 460, 260, 70, 20, _S3 ),
    wccreate( (e_szerkezet,11000), static, 'Héj', 130, 260, 30, 20, _S4 ),
    wccreate( (e_szerkezet,11001), static, 'Alhéj', 170, 260, 30, 20, _S4 ),
    wccreate( (e_szerkezet,11002), static, 'Elektronok száma', 210, 260, 90, 20, _S4 ),
    wccreate( (e_szerkezet,4000), listbox, 'List1', 140, 280, 20, 260, _S5 ),
    wccreate( (e_szerkezet,4001), listbox, 'List2', 180, 280, 20, 260, _S5 ),
    wccreate( (e_szerkezet,4002), listbox, 'List3', 240, 280, 20, 260, _S5 ),
    wccreate( (e_szerkezet,1002), button, 'Kivételes elemek', 400, 370, 110, 30, _S2 ),
    wccreate( (e_szerkezet,11003), static, 'Kérem az elem rendszámát:', 310, 260, 130,
20, _S4 ),
    wccreate( (e_szerkezet,9000), rich, '', 10, 10, 560, 240, _S6 ).

kérdés6:- e_szerkezet,
        absolute_file_name( examples('elektronszerkezet.rtf'), File8 ),
        wrchlod((e_szerkezet,9000),1,File8),
        window_handler( e_szerkezet, e_szerkezet_handler ),
        call_dialog(e_szerkezet, _).

e_szerkezet_handler( e_szerkezet, msg_close, _, done ).
e_szerkezet_handler( (e_szerkezet,1001), msg_button, _, done ).
e_szerkezet_handler( (e_szerkezet,1002), msg_button, _, _):- kérdés7.
e_szerkezet_handler( (e_szerkezet,1000), msg_button, _, _):-
wtext((e_szerkezet,8000),Pos), number_string(E,Pos), N is 1, L is 0, Osszeg is 1,
elektronpalya(E,N,L,Osszeg), új3.
új3:- újra3, window_handler( újra3, újra3_handler ), call_dialog(újra3, _).
újra3_handler( újra3,1000), msg_button, _, _):- kérdés6.
újra3_handler( újra3,1001), msg_button, _, _):-
        wclose(újra3),
        wclose(e_szerkezet),
        kérdés1.

```

újra3 :-

```
_S1 = [ws_caption,ws_sysmenu,dlg_ownedbyprolog],
_S2 = [ws_child,ws_visible,ss_center],
_S3 =
[ws_child,ws_tabstop,ws_visible,bs_pushbutton,bs_text,bs_center,bs_vcenter],
wcreate( újra3, 'Visszalép', 291, 112, 296, 125, _S1 ),
wcreate( újra3,11000), static, 'Folytatja más adatokkal?', 100, 10, 130, 20,
_S2 ),
wcreate( újra3,1000), button, 'Igen', 20, 50, 110, 30, _S3 ),
wcreate( újra3,1001), button, 'Nem', 160, 50, 110, 30, _S3 ).
```

```
elektronpálya(E,N,L,Osszeg):- E>0, N<8, Osszeg is N+L, darab(L,E,Eb,E3),
number_string(N, N3), wlstadd((e_szerkezet,4000),-1,N3,0),
number_string(L, L3), wlstadd((e_szerkezet,4001),-1,L3,0),
number_string(Eb, Eb3), wlstadd((e_szerkezet,4002),-1,Eb3,0),
változtat(N,L,Osszeg,N2,L2,Osszeg2), !,
elektronpálya(E3,N2,L2,Osszeg2).
```

elektronpálya(E,N,L,Osszeg):- N=8, L>=0, Osszeg>0, E>0.

elektronpálya(E,N,L,Osszeg):- N>0, L>=0, Osszeg>0, E=0.

változtat(N,L,Osszeg,N2,L2,Osszeg2):- Osszeg=N, L=0, Osszeg2 is Osszeg+1,  
L2 is N // 2, N2 is Osszeg2-L2.

változtat(N,L,Osszeg,N2,L2,Osszeg2):- Osszeg>N, Osszeg2 is Osszeg, N2 is  
N+1,

L2 is L-1.

darab(L,E,Eb,E3):- Db is 4\*L+2, E>=Db, Eb is Db, E3 is E-Db.

darab(L,E,Eb,E3):- Db is 4\*L+2, E<Db, Eb is E, E3 is 0.

kivételek :- hangol5,

```
_S1 = [ws_caption,ws_sysmenu,dlg_ownedbyprolog],
_S2 =
[ws_child,ws_disabled,ws_border,ws_tabstop,ws_visible,es_left,es_multiline,es_a
utohscroll,es_autovscroll],
_S3 = [ws_child,ws_border,ws_visible],
wcreate( kivételek, 'Kivételes elemek', 182, 57, 736, 605, _S1 ),
wcreate( kivételek,9000), rich, '', 10, 370, 710, 200, _S2 ),
wcreate( kivételek,10000), grafix, 'Grafix1', 10, 10, 710, 350, _S3 ).
```

hangol5:-



```

absolute_file_name( examples('kivételek2.bmp'), File ),
gfx_bitmap_load( rajz5, File ), !.

kérdés7:- kivételek,
        absolute_file_name( examples('kivételek szöveg.rtf'), File5 ),
        wrchlod((kivételek,9000),1,File5),
        window_handler( kivételek, kivételek_handler ),
        call_dialog(kivételek, _).

kivételek_handler( kivételek, msg_close, _, done ).
kivételek_handler( (kivételek,10000), msg_paint, grafix, _ ) :-
    gfx_paint( (kivételek,10000) ),
    gfx( bitmap(0,0,2000,2000,0,0,rajz5) ),
    gfx_end( (kivételek,10000) ).

oxidációs :-
    _S1 = [ws_caption,ws_sysmenu,dlg_ownedbyprolog],
    _S2 =
[ws_child,ws_border,ws_disabled,ws_tabstop,ws_visible,es_left,es_multiline,es_a
utohscroll,es_autovscroll],
    _S3 =
[ws_child,ws_border,ws_tabstop,ws_visible,es_left,es_multiline,es_autohscroll,es
_autovscroll],
    _S4 = [ws_child,ws_visible,ss_left],
    _S5 = [ws_child,ws_visible,ws_border,ws_tabstop,ws_vscroll,lbs_sort],
    _S6 =
[ws_child,ws_tabstop,ws_visible,bs_pushbutton,bs_text,bs_center,bs_vcenter],
    _S7 = [ws_child,ws_tabstop,ws_visible,cbs_dropdownlist],
    _S8 = [ws_child,ws_border,ws_tabstop,ws_visible,ws_vscroll,lbs_sort],
    _S9 =
[ws_child,ws_border,ws_tabstop,ws_visible,es_right,es_multiline,es_autohscroll,
es_autovscroll],
    wdcreate( oxidációs, 'Oxidációs számok', 232, 59, 676, 495, _S1 ),
    wccreate( (oxidációs,9000), rich, '', 10, 230, 510, 230, _S2 ),
    wccreate( (oxidációs,8000), edit, '', 190, 40, 50, 20, _S3 ),
    wccreate( (oxidációs,11000), static, 'Kérem az elem rendszámát:', 40, 40,
140, 20, _S4 ),
    wccreate( (oxidációs,11001), static, 'Megadott elem oxidációs számai', 10, 10,
170, 20, _S4 ),

```

```

wcreate( (oxidációs,11002), static, 'Az elem neve:', 40, 100, 80, 20, _S4 ),
wcreate( (oxidációs,11003), static, 'Jellemző oxidációs számai:', 40, 70, 140,
20, _S4 ),
wcreate( (oxidációs,4000), listbox, 'List1', 190, 70, 20, 150, _S5 ),
wcreate( (oxidációs,1000), button, 'megjelenít', 250, 20, 70, 30, _S6 ),
wcreate( (oxidációs,11004), static, 'Megadott csoport oxidációs számai', 340,
10, 180, 20, _S4 ),
wcreate( (oxidációs,11005), static, 'Kérem a csoport kiválasztását:', 370, 40, 150,
20, _S4 ),
wcreate( (oxidációs,5000), combobox, 'Combo1', 530, 40, 60, 150, _S7 ),
wcreate( (oxidációs,5001), combobox, 'Combo2', 600, 40, 60, 80, _S7 ),
wcreate( (oxidációs,1001), button, 'számol', 590, 80, 70, 30, _S6 ),
wcreate( (oxidációs,11006), static, 'Jellemző oxidációs számok:', 370, 70,
150, 20, _S4 ),
wcreate( (oxidációs,4001), listbox, 'List2', 530, 70, 20, 340, _S8 ),
wcreate( (oxidációs,8001), edit, '', 40, 130, 110, 20, _S9 ).

kérdés8:- oxidációs ,

```

```

absolute_file_name( examples('oxidációs számok.rtf'), File5 ),
wrchld((oxidációs,9000),1,File5),
wcbadd((oxidációs,5000),-1,'1',0),
wcbadd((oxidációs,5000),-1,'2',0),
wcbadd((oxidációs,5000),-1,'3',0),
wcbadd((oxidációs,5000),-1,'4',0),
wcbadd((oxidációs,5000),-1,'5',0),
wcbadd((oxidációs,5000),-1,'6',0),
wcbadd((oxidációs,5000),-1,'7',0),
wcbadd((oxidációs,5000),-1,'8',0),
wcmbsel((oxidációs,5000),0,0),
wcbadd((oxidációs,5001),-1,'a',0),
wcbadd((oxidációs,5001),-1,'b',0),
wcmbsel((oxidációs,5001),0,0),
window_handler( oxidációs , oxidációs_handler ),
call_dialog(oxidációs , _).

```

```

oxidációs_handler( oxidációs , msg_close, _, done ).

```

```

oxidációs_handler( (oxidációs,1000), msg_button, _, _):-

```

wtext((oxidációs,8000),Pos), number\_string(E,Pos),  
 elem(E,X,\_,\_,\_), wtext((oxidációs,8001),X), oxidációs\_száma(X,C),  
 number\_string(C,Szoveg), wlstadd((oxidációs,4000),-1, Szoveg,0), fail.

oxidációs\_handler( (oxidációs,1001), msg\_button, \_, \_):-

wlbgxget((oxidációs,5000),Pos), number\_string(E,Pos),  
 wlbgxget((oxidációs,5001),E2), !,

csoport\_számai(E,E2,C), number\_string(C,Szoveg),  
 wlstadd((oxidációs,4001),-1, Szoveg,0), fail.

elem(1,'hidrogén',1,1,'a',2.1). elem(2,'hélium',1,8,'a',10).  
 elem(3,'lítium',2,1,'a',1.0). elem(4,'berillium',2,2,'a',1.5).  
 elem(5,'bór',2,3,'a',2.0). elem(6,'szén',2,4,'a',2.5). elem(7,'nitrogén',2,5,'a',3.0).  
 elem(8,'oxigén',2,6,'a',3.5). elem(9,'fluor',2,7,'a',4.0).  
 elem(10,'neon',2,8,'a',10). elem(11,'nátrium',3,1,'a',0.9).  
 elem(12,'magnézium',3,2,'a',1.2).  
 elem(13,'alumínium',3,3,'a',1.5). elem(14,'szilícium',3,4,'a',1.8).  
 elem(15,'foszfor',3,5,'a',2.1). elem(16,'kén',3,6,'a',2.5).  
 elem(17,'klór',3,7,'a',3.0). elem(18,'argon',3,8,'a',10).  
 elem(19,'kálium',4,1,'a',0.8). elem(20,'kalcium',4,2,'a',1.0).  
 elem(21,'szkandium',4,3,'b',1.3). elem(22,'titán',4,4,'b',1.6).  
 elem(23,'vanádium',4,5,'b',1.6). elem(24,'króm',4,6,'b',1.6).  
 elem(25,'mangán',4,7,'b',1.5). elem(26,'vas',4,8,'b',1.8).  
 elem(27,'kobalt',4,8,'b',1.8). elem(28,'nikkel',4,8,'b',1.8).  
 elem(29,'réz',4,1,'b',1.9). elem(30,'cink',4,2,'b',1.6).  
 elem(31,'gallium',4,3,'a',1.6). elem(32,'germánium',4,4,'a',1.8).  
 elem(33,'arzén',4,5,'a',2.0). elem(34,'szelén',4,6,'a',2.4).  
 elem(35,'bróm',4,7,'a',2.8). elem(36,'kripton',4,8,'a',10).  
 elem(37,'rubídium',5,1,'a',0.7). elem(38,'stroncium',5,2,'a',0.9).  
 elem(39,'ittrium',5,3,'b',1.3). elem(40,'cirkónium',5,4,'b',1.4).  
 elem(41,'nióbium',5,5,'b',1.6). elem(42,'molibdén',5,6,'b',1.8).  
 elem(43,'technécium',5,7,'b',1.9). elem(44,'ruténium',5,8,'b',2.2).  
 elem(45,'ródiium',5,8,'b',2.2). elem(46,'palládium',5,8,'b',2.2).

elem(47,`ezüst`,5,1,`b`,1.9). elem(48,`kadmium`,5,2,`b`,1.7).  
 elem(49,`indium`,5,3,`a`,1.7). elem(50,`ón`,5,4,`a`,1.8).  
 elem(51,`antimon`,5,5,`a`,1.9). elem(52,`tellúr`,5,6,`a`,2.1).  
 elem(53,`jód`,5,7,`a`,2.5). elem(54,`xenon`,5,8,`a`,10).  
 elem(55,`cézium`,6,1,`a`,0.7). elem(56,`bárium`,6,2,`a`,0.9).  
 elem(57,`lantán`,6,3,`b`,1.1). elem(72,`hafnium`,6,4,`b`,1.3).  
 elem(73,`tantál`,6,5,`b`,1.5). elem(74,`volfram`,6,6,`b`,1.7).  
 elem(75,`rénium`,6,7,`b`,1.9). elem(76,`ozmium`,6,8,`b`,2.2).  
 elem(77,`irídium`,6,8,`b`,2.2). elem(78,`platina`,6,8,`b`,2.2).  
 elem(79,`arany`,6,1,`b`,2.4). elem(80,`higany`,6,2,`b`,1.9).  
 elem(81,`tallium`,6,3,`a`,1.8). elem(82,`ólom`,6,4,`a`,1.8).  
 elem(83,`bizmut`,6,5,`a`,1.9). elem(84,`polónium`,6,6,`a`,2.0).  
 elem(86,`radon`,6,8,`a`,10).  
 elem(87,`francium`,7,1,`a`,0.7). elem(88,`rádium`,7,2,`a`,1.1).  
 elem(89,`aktínium`,7,3,`b`,1.1).  
 oxidációs\_szám(X,0):- elem(\_X,\_,\_,\_).  
 oxidációs\_szám(X,Y):- elem(\_X,\_,Y,\_,\_), Y<6.  
 oxidációs\_szám(X,Y):- elem(\_X,\_,Y,`b`,\_), Y>5, Y<8.  
 oxidációs\_szám(X,3):- elem(\_X,\_,5,`a`,\_).  
 oxidációs\_szám(X,-3):- elem(\_X,S,5,`a`,\_), S<6.  
 oxidációs\_szám(X,-2):- elem(\_X,S,6,`a`,\_), S<6.  
 oxidációs\_szám(X,4):- elem(\_X,S,6,`a`,\_), S>2, S<7.  
 oxidációs\_szám(X,-1):- elem(\_X,\_,7,`a`,\_).  
 oxidációs\_szám(X,1):- elem(\_X,S,7,`a`,\_), S>2, S<7.  
 oxidációs\_szám(X,5):- elem(\_X,S,7,`a`,\_), S>2, S<7.  
 oxidációs\_szám(X,3):- elem(\_X,4,O,`b`,\_), O>3.  
 oxidációs\_szám(X,2):- elem(\_X,4,O,`b`,\_), O>4.  
 oxidációs\_szám(X,2):- elem(\_X,S,8,`b`,\_), S>4, S<7.

oxidációs\_száma(X,4):- elem(\_X,S,8,'b',\_), S>4, S<7.

oxidációs\_száma('tallium',1). oxidációs\_száma('szén',-4).

oxidációs\_száma('szén',2).

oxidációs\_száma('ón',2). oxidációs\_száma('ólom',2). oxidációs\_száma('nitrogén',4).

oxidációs\_száma('nitrogén',2).

oxidációs\_száma('foszfor',4). oxidációs\_száma('oxigén',-1).

oxidációs\_száma('kén',2). oxidációs\_száma('kén',6).

oxidációs\_száma('szelén',6). oxidációs\_száma('tellúr',6).

oxidációs\_száma('polónium',2).

oxidációs\_száma('klór',3). oxidációs\_száma('klór',7). oxidációs\_száma('jód',7).

oxidációs\_száma('réz',2). oxidációs\_száma('cink',1). oxidációs\_száma('arany',3).

oxidációs\_száma('higany',1).

oxidációs\_száma('vanádium',4). oxidációs\_száma('niobium',3).

oxidációs\_száma('mangán',4). oxidációs\_száma('mangán',6).

oxidációs\_száma('molibdén',2). oxidációs\_száma('molibdén',3).

oxidációs\_száma('molibdén',4). oxidációs\_száma('molibdén',5).

oxidációs\_száma('volfram',2). oxidációs\_száma('volfram',3).

oxidációs\_száma('volfram',4). oxidációs\_száma('volfram',5).

oxidációs\_száma('ruténium',3). oxidációs\_száma('ruténium',6).

oxidációs\_száma('ruténium',8).

oxidációs\_száma('ródiium',3). oxidációs\_száma('rénium',6).

oxidációs\_száma('rénium',4).

oxidációs\_száma('rénium',2). oxidációs\_száma('rénium',-1).

oxidációs\_száma('ozmium',3). oxidációs\_száma('ozmium',6).

oxidációs\_száma('ozmium',8). oxidációs\_száma('irídium',3).

oxidációs\_száma('irídium',6).

vegyület([],[], Osszeg):- Osszeg=0.

vegyület([A|Elemlista],[T|Mennyiseglista], Osszeg):- oxidációs\_száma(A,O),  
not(O=0) , Osszeg2 is Osszeg+O\*T,

vegyület(Elemlista,Mennyiseglista, Osszeg2).

csoport\_számai(A,B,C):- elem(\_X,\_A,B,\_), oxidációs\_száma(X,C).

hasonlit\_en(A,B):- elem(\_A,\_\_,\_,X), elem(\_B,\_\_,\_,Y), X>Y, write(A), nl.

hasonlit\_en(A,B):- elem(\_A,\_\_,\_,X), elem(\_B,\_\_,\_,Y), X=Y,  
write(' egyforma'), nl.

hasonlit\_en(A,B):- elem(\_A,\_\_,\_,X), elem(\_B,\_\_,\_,Y), X<Y, write(B), nl.

elektronegativitás(X,Y):-elem(\_X,\_\_,\_,Y).

en\_csoport(P,Q,R):-elem(\_\_,\_,P,Q,R).

en\_periódus(S,En):-elem(\_\_,S,\_\_,En).

negatív :-

\_S1 = [ws\_caption,ws\_sysmenu,dlg\_ownedbyprolog],

\_S2 =

[ws\_child,ws\_border,ws\_disabled,ws\_tabstop,ws\_visible,es\_left,es\_multiline,es\_a  
utohscroll,es\_autovscroll],

\_S3 =

[ws\_child,ws\_border,ws\_tabstop,ws\_visible,es\_right,es\_multiline,es\_autohscroll,  
es\_autovscroll],

\_S4 = [ws\_child,ws\_visible,ss\_left],

\_S5 =

[ws\_child,ws\_tabstop,ws\_visible,bs\_pushbutton,bs\_text,bs\_center,bs\_vcenter],

\_S6 = [ws\_child,ws\_tabstop,ws\_visible,cbs\_dropdownlist],

\_S7 = [ws\_child,ws\_border,ws\_tabstop,ws\_visible,ws\_vscroll],

wdcreate( negatív, 'Elektronegativitás', 232, 59, 676, 595, \_S1 ),

wccreate( (negatív,9000), rich, '', 10, 160, 350, 230, \_S2 ),

wccreate( (negatív,8000), edit, '', 190, 40, 50, 20, \_S3 ),

wccreate( (negatív,11000), static, `Kérem az elem rendszámát:`, 40, 40, 140,  
 20, \_S4 ),  
 wccreate( (negatív,11001), static, `Megadott elem elektronegativitása`, 10, 10, 170,  
 20, \_S4 ),  
 wccreate( (negatív,11002), static, `Az elem neve:`, 40, 70, 80, 20, \_S4 ),  
 wccreate( (negatív,11003), static, `Elektronegativitás értéke:`, 40, 100, 140,  
 20, \_S4 ),  
 wccreate( (negatív,1000), button, `megjelenít`, 250, 40, 70, 30, \_S5 ),  
 wccreate( (negatív,11004), static, `Megadott csoport, periódus  
 elektronegativitás értékei`, 340, 10, 270, 20, \_S4 ),  
 wccreate( (negatív,11005), static, `Kérem a csoport kiválasztását:`, 370, 40,  
 150, 20, \_S4 ),  
 wccreate( (negatív,5000), combobox, `Combo1`, 530, 40, 60, 150, \_S6 ),  
 wccreate( (negatív,5001), combobox, `Combo2`, 600, 40, 60, 80, \_S6 ),  
 wccreate( (negatív,1001), button, `számol`, 590, 80, 70, 30, \_S5 ),  
 wccreate( (negatív,11006), static, `Elektronegativitás értékei:`, 370, 70, 150,  
 20, \_S4 ),  
 wccreate( (negatív,4001), listbox, `List2`, 530, 70, 20, 150, \_S7 ),  
 wccreate( (negatív,8001), edit, ``, 130, 70, 110, 20, \_S3 ),  
 wccreate( (negatív,8002), edit, ``, 190, 100, 50, 20, \_S3 ),  
 wccreate( (negatív,11007), static, `Kérem a periódus kiválasztását:`, 370, 230,  
 150, 20, \_S4 ),  
 wccreate( (negatív,11008), static, `Elektronegativitás értékei:`, 370, 260, 150,  
 20, \_S4 ),  
 wccreate( (negatív,5002), combobox, `Combo3`, 530, 230, 60, 130, \_S6 ),  
 wccreate( (negatív,4000), listbox, `List1`, 610, 230, 20, 310, \_S7 ),  
 wccreate( (negatív,1002), button, `számol`, 530, 260, 70, 30, \_S5 ).

kérdés9:- negatív,

```
absolute_file_name( examples('elektronegativitas.rtf'), File5 ),  
wrchlod((negatív,9000),1,File5),  
wcbmadd((negatív,5000),-1,`1`,0), wcbmadd((negatív,5000),-  
1,`2`,0),  
wcbmadd((negatív,5000),-1,`3`,0), wcbmadd((negatív,5000),-  
1,`4`,0),  
wcbmadd((negatív,5000),-1,`5`,0), wcbmadd((negatív,5000),-  
1,`6`,0),  
wcbmadd((negatív,5000),-1,`7`,0), wcbmadd((negatív,5000),-  
1,`8`,0),  
wcmbsel((negatív,5000),0,0), wcbmadd((negatív,5001),-1,`a`,0),  
wcbmadd((negatív,5001),-1,`b`,0), wcmbsel((negatív,5001),0,0),  
wcbmadd((negatív,5002),-1,`1`,0), wcbmadd((negatív,5002),-  
1,`2`,0),  
wcbmadd((negatív,5002),-1,`3`,0), wcbmadd((negatív,5002),-  
1,`4`,0),  
wcbmadd((negatív,5002),-1,`5`,0), wcbmadd((negatív,5002),-  
1,`6`,0),  
wcbmadd((negatív,5002),-1,`7`,0), wcmbsel((negatív,5002),0,0),  
window_handler( negatív, negatív_handler ),  
call_dialog(negatív, _).  
negatív_handler( negatív, msg_close, __, done ).  
negatív_handler( (negatív,1000), msg_button, __, _):-  
wtext((negatív,8000),Pos), number_string(E,Pos),  
elem(E,X,_,_,_), wtext((negatív,8001),X),elektronegativitas(X,Y),  
number_string(Y,Sz), wtext((negatív,8002),Sz).
```



```

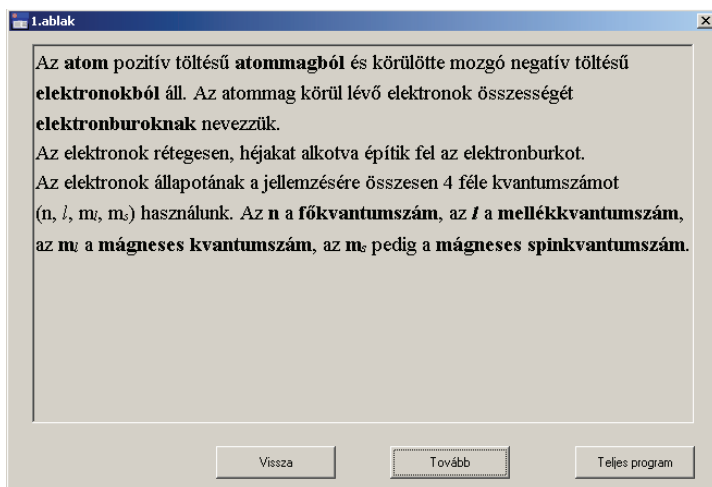
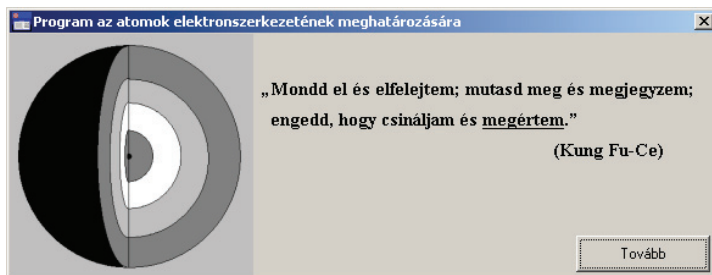
negatív_handler( (negatív,1001), msg_button, _, _):-
    wlbxget((negatív,5000),Pos), number_string(E,Pos),
    wlbxget((negatív,5001),Q),
    en_csoport(E,Q,R), number_string(R,Szoveg), wlstadd((negatív,4001),-1,
    Szoveg,0), fail.

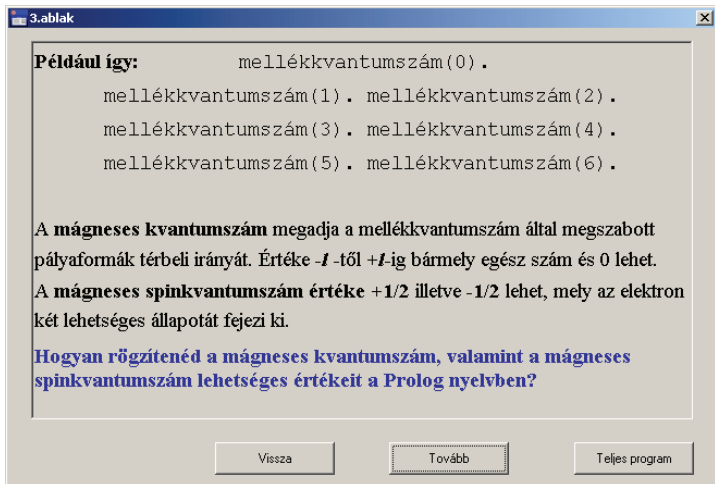
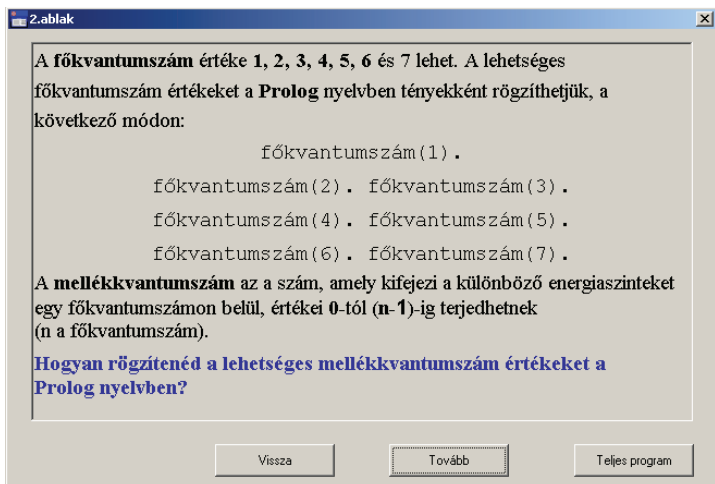
negatív_handler( (negatív,1002), msg_button, _, _):-
    wlbxget((negatív,5002),Pos), number_string(E,Pos), en_periódus(E,En),
    number_string(En,Szoveg), wlstadd((negatív,4000),-1, Szoveg,0), fail.

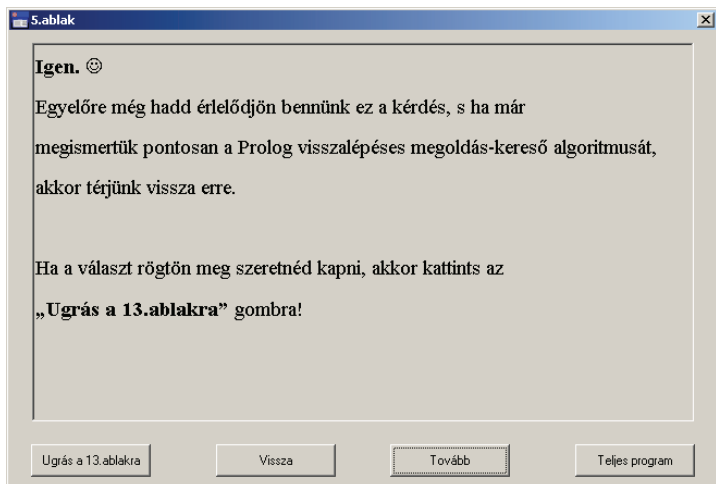
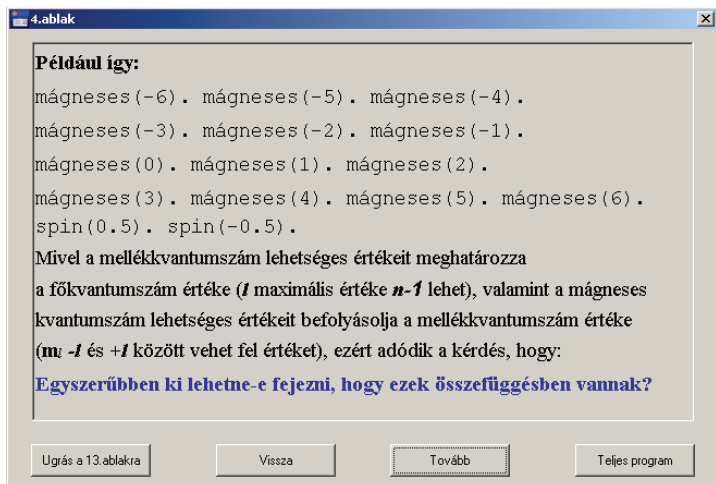
```

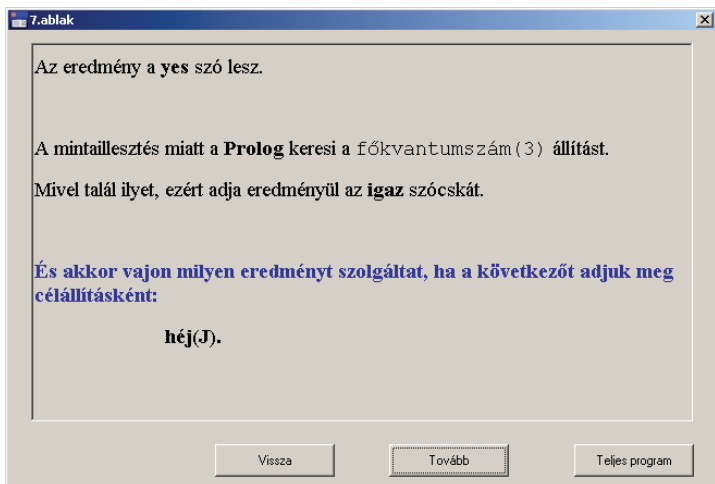
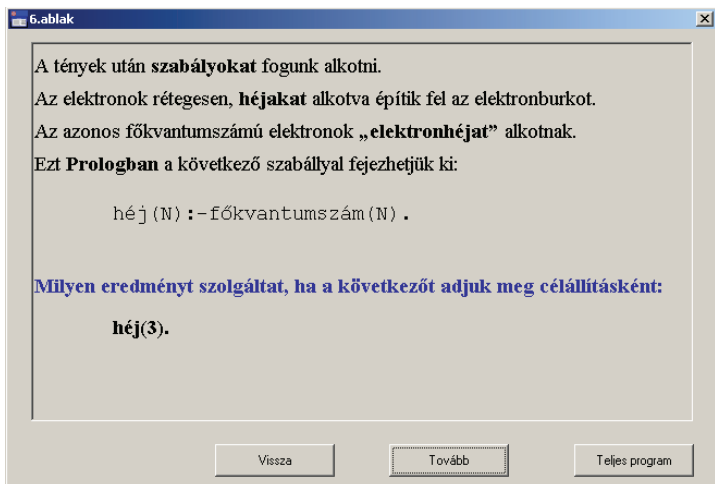
## B függelék -

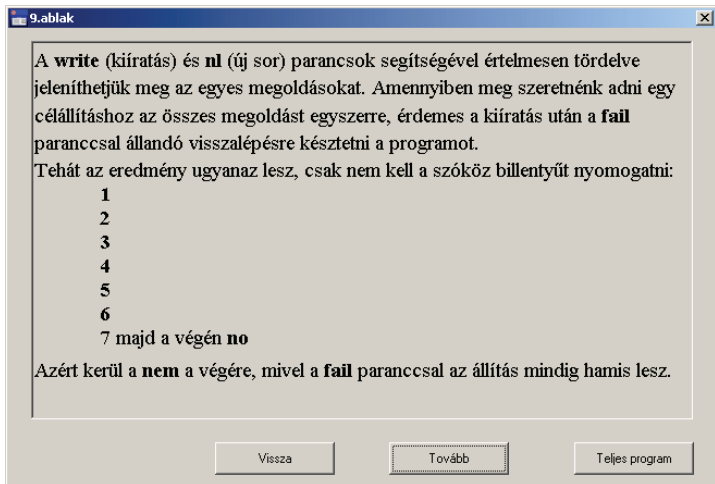
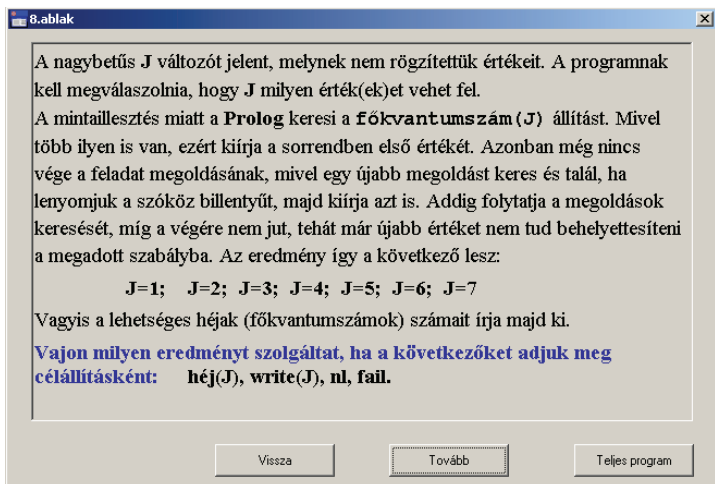
### Programozott tananyag az atomok elektronszerkezetének meghatározására

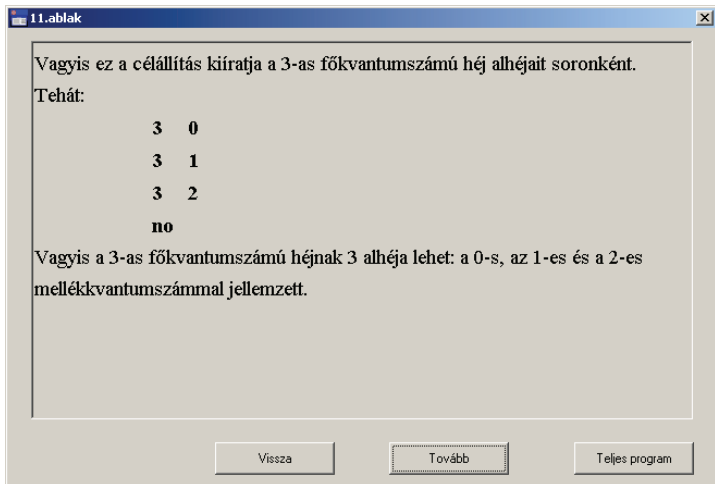
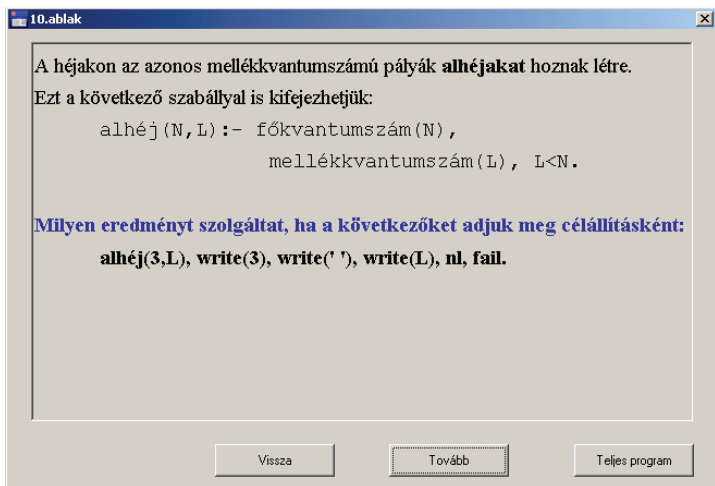


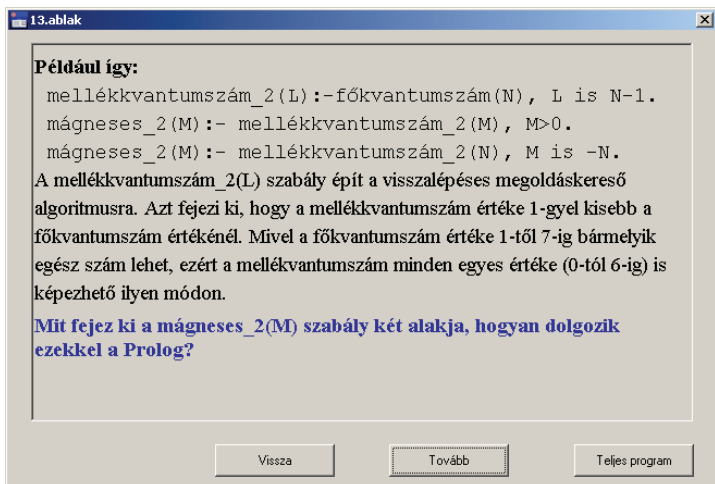
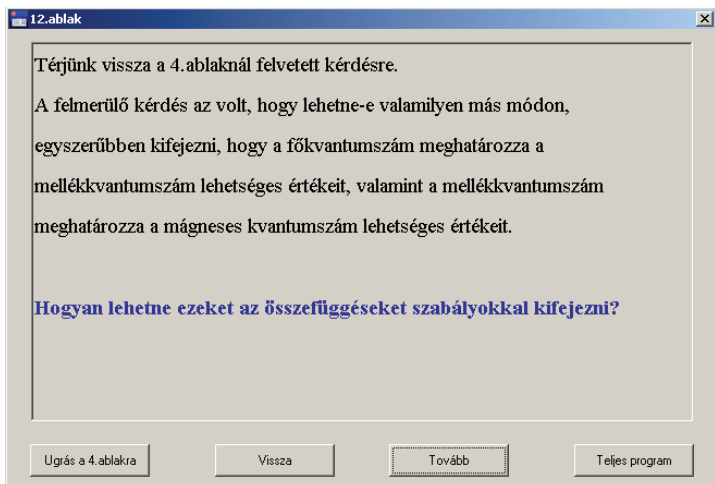




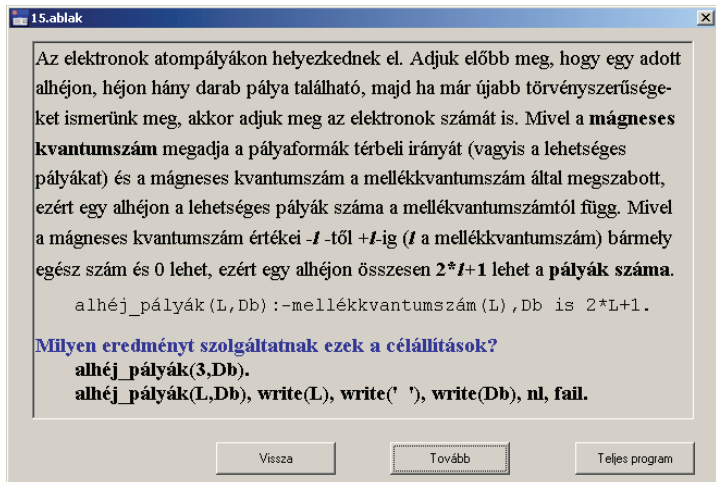
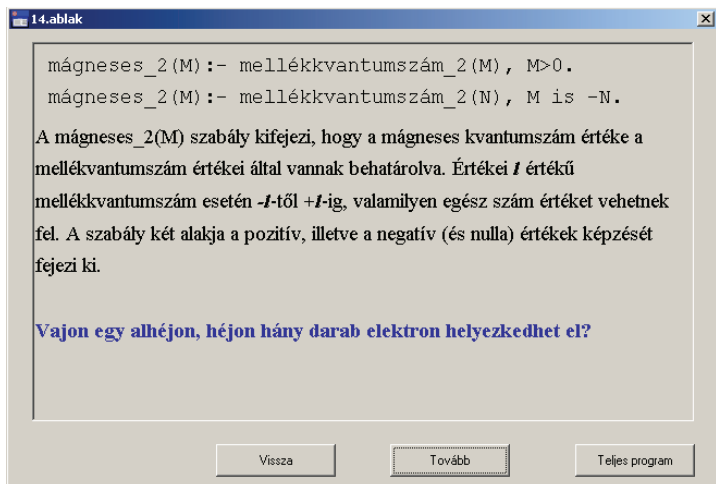












16.ablak

Az **alhěj\_pályák(3,Db)** célállítás kiírja a 3-as mellékkvantumszámú alhéj pályáinak a számát. Tehát: **Db=7** (vagyis a 3-as alhéj 7 pályával rendelkezik).

Az **alhěj\_pályák(L,Db), write(L), write(' '), write(Db), nl, fail.** állítás eredménye:

```

0 1
1 3
2 5
3 7
4 9
5 11
6 13 és végül no

```

Vajon egy héjon hány darab pálya helyezkedhet el?

Vissza      Tovább      Teljes program

17.ablak

Az **n.** héjon a pályák maximális száma  **$n^2$**  (**n** a főkvantumszám).

Ezt a következő szabállyal is kifejezhetjük:

$$\text{héj\_pályák}(N, Db) :- \text{főkvantumszám}(N), \\ Db \text{ is } N*N.$$

Milyen eredményt szolgáltatnak ezek a célállítások?

**héj\_pályák(3,Db).**

**héj\_pályák(N,Db), write(N), write(' '),write(Db), nl, fail.**

Vissza      Tovább      Teljes program

18.ablak

A **hég\_pályák(3,Db)** célállítást kiírja a 3-as főkvantumszámú hég pályáinak a számát. Tehát: **Db=9** (vagyis a 3-as hég 9 pályával rendelkezik).

A **hég\_pályák(N,Db)**, **write(N)**, **write(' ')**, **write(Db)**, **nl**, **fail.** állítás eredménye:

```

1 1
2 4
3 9
4 16
5 25
6 36
7 49 és végül no

```

Vajon egy alhéjon, héjon hány darab elektron helyezkedhet el?

Vissza      Tovább      Teljes program

19.ablak

Az elektronok számának a megadásában segítségünkre van a **Pauli-féle tilalmi elv.** Eszerint valamely atomban két vagy több elektronnak sohasem lehet mind a négy kvantumszáma ( $n, l, m_l, m_s$ ) azonos. Mivel a mágneses spinquantumszám értéke csak kétféle lehet ( $-0.5$  ill.  $+0.5$ ), ebből az következik, hogy egy atom-pályán csak 2 elektron helyezkedhet el. Vagyis az egy alhéjon lévő elektronok száma  $4 \cdot l + 2$ . Az  $n$ . héjon pedig az elektronok maximális száma  $2n^2$

Ezt a következő szabályokkal is kifejezhetjük:

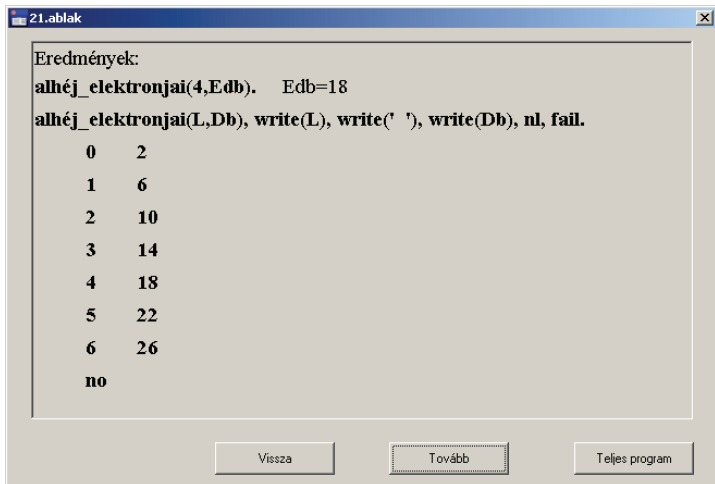
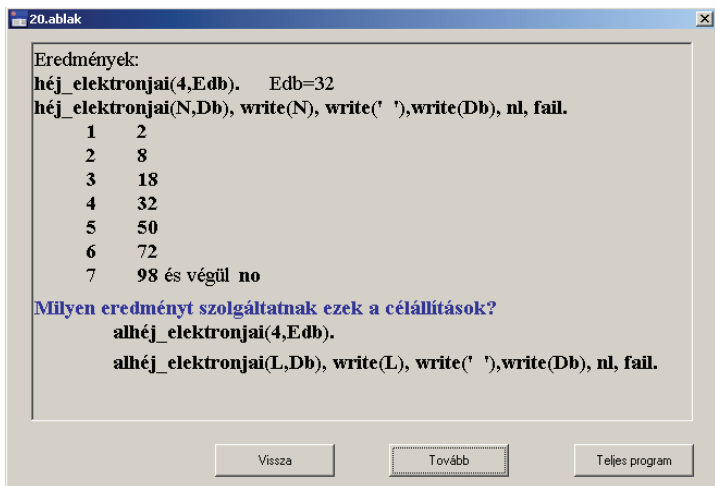
hég\_elektronjai(N,Edb): -főkvantumszám(N), Edb is  $2 \cdot N \cdot N$ .  
 alhég\_elektronjai(L,Edb): -mellékkvantumszám(L), Edb is  $4 \cdot L + 2$ .

Milyen eredményt szolgáltatnak ezek a célállítások?

**hég\_elektronjai(4,Edb).**

**hég\_elektronjai(N,Db), write(N), write(' '),write(Db), nl, fail.**

Vissza      Tovább      Teljes program





24.ablak

Eredmény:

0	0	0.5
0	0	-0.5
1	-1	0.5
1	-1	-0.5
1	0	0.5
1	0	-0.5
1	1	0.5
1	1	-0.5 és végül

no

Mivel  $N=2$ , így  $L$  0, illetve 1 értéket vehet fel.  $M$  lehetséges értékei 0, illetve -1, 0 és 1 lehet. A spinquantumszám pedig -0.5 és +0.5 értékeket vehet fel.

Milyen eredményt szolgáltat ez a célállítás?

kvantumok( $N,L,M,S$ ), write( $N$ ), write(' '), write( $L$ ), write(' '), write( $M$ ),  
write(' '), write( $S$ ), nl, fail.

Vissza

Tovább

Teljes program

25.ablak

Ez a célállítás kiírja eredményül az összes ( $n=7$ -ig) lehetséges kvantumszám-kombinációt (ez összesen 280-félét jelent):

1	0	0	0.5
1	0	0	-0.5
2	0	0	0.5
2	0	0	-0.5
...			...
7	6	6	0.5
7	6	6	-0.5 és végül

no

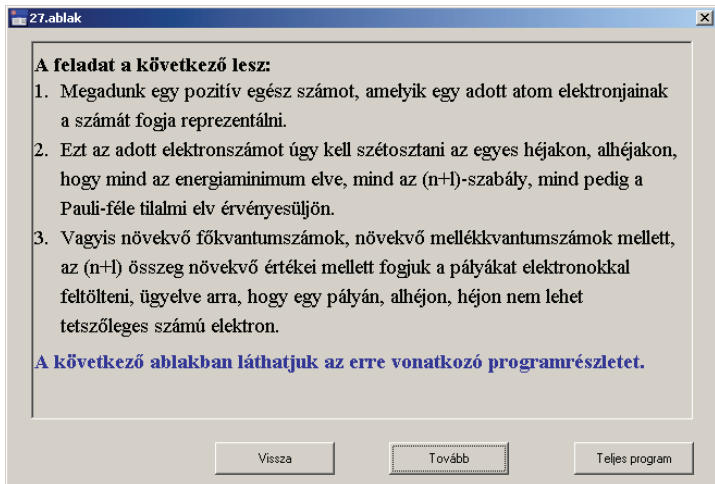
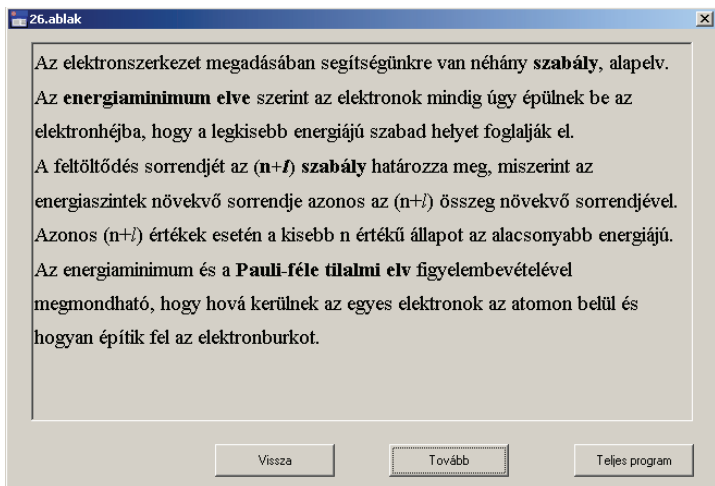
Ezek alapján próbáljuk meg kitalálni, hogy milyen egy adott elem (atom) elektronkonfigurációja?

(Vagyis, hogy az adott elem atomjának elektronjai milyen pályákat töltenek be, illetve, hogy melyik elektronpályán mennyi elektron található.)

Vissza

Tovább

Teljes program



28.ablak

```

elektronpálya(E,N,L):- E>0, N<8, darab(L,E,Eb,E3),
                        write(N), write(' '),
                        write(L), write(' '),
                        write(Eb), nl,
                        változtat(N,L,N2,L2), !,
                        elektronpálya(E3,N2,L2).

elektronpálya(_,N,_):- N=8.
elektronpálya(E,_,_):- E=0.

darab(L,E,Eb,E3):-Db is 4*L+2,E>=Db, Eb is Db, E3 is E-Db.
darab(L,E,Eb,E3):-Db is 4*L+2,E<Db, Eb is E, E3 is 0.

változtat(N,L,N2,L2):- L=0, Osszeg2 is N+1,
                        L2 is N // 2, N2 is Osszeg2-L2.
változtat(N,L,N2,L2):- L>0, N2 is N+1, L2 is L-1.

```

Az **elektronpálya** szabály felhasználja a **darab**, és a **változtat** szabályokat is.

Vissza

Tovább

Teljes program

29.ablak

```

elektronpálya(E,N,L):- E>0, N<8, darab(L,E,Eb,E3),
                        write(N),write(' '),write(L),write(' '),write(Eb),nl,
                        változtat(N,L,N2,L2), !, elektronpálya(E3,N2,L2).
elektronpálya(_,N,_):- N=8.
elektronpálya(E,_,_):- E=0.

```

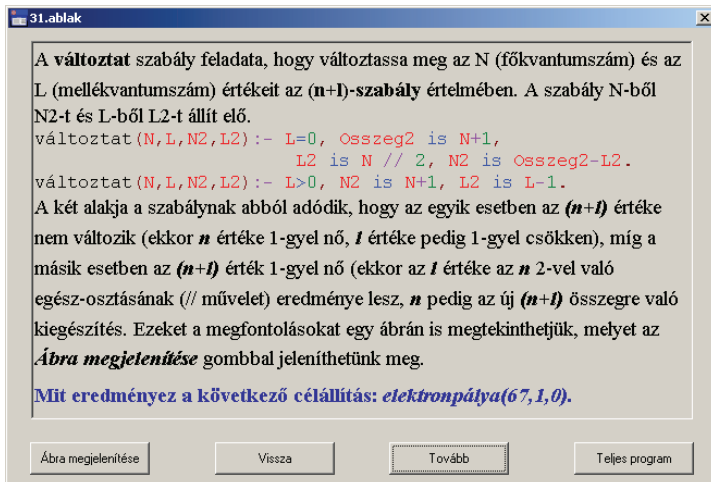
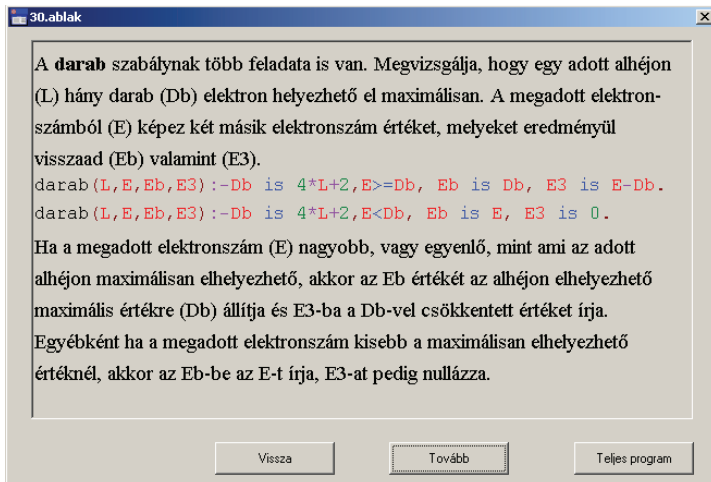
Az **elektronpálya** szabály egy megadott elektronszám (E) esetén az N (fő) és L (mellék) kvantumszámok szerint, az **(n+l)-szabály** alapján írja ki az egyes alhéjakon (N, L) található elektronok számát. Az első híváskor az N=1, az L=0 legyen, mivel ez az első alhéj, amelyik kiépül. A szabálynak 3 alakja is van. Az első szabály végén ismét láthatjuk az elektronpálya szabály hívását, vagyis ez a rész rekurzív hívást tartalmaz. A másik két **elektronpálya** szabály a rekurzív lezárásáért felel. Vagyis N=8-nál (ha „elfogytak” a héjak), illetve E=0-nál (ha „elfogytak” az elektronok) áll meg a rekurzió.

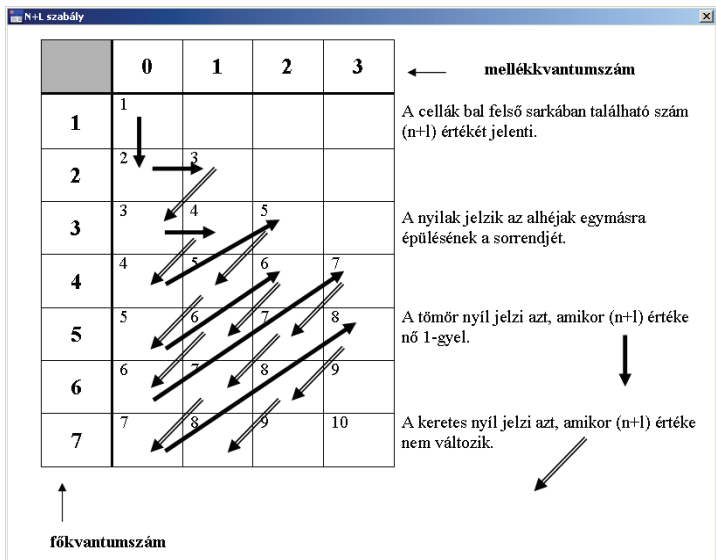
Vissza

Tovább

Teljes program







**32.ablak**

Az **elektronpálya**(67,1,0) célállítás eredménye:

1	0	2
2	0	2
2	1	6
3	0	2
3	1	6
4	0	2
3	2	10
4	1	6
5	0	2
4	2	10
5	1	6
6	0	2
4	3	11

Vissza      Tovább      Teljes program

33.ablak

Ellenőrizd az  $(n+l)$  értékeket! Valóban igaz rájuk az  $(n+l)$  szabály?

n	l	$(n+l)$ összeg
1	0	1
2	0	2
2	1	3
3	0	3
3	1	4
4	0	4
3	2	5
4	1	5
5	0	5
4	2	6
5	1	6
6	0	6
4	3	7

Vissza      Tovább      Teljes program

34.ablak

**A program véget ért. Remélem amellet, hogy sikerült elsajátítanod az ismereteket, érdekesnek is találtad. Ellenőrizd tudásodat a következő kérdések segítségével:**

- Milyen kvantumszámok és milyen kvantumszámú héjak léteznek?
- Egy adott héjnak milyen alhéjai lehetnek?
- Az adott héjnak hány darab elektrópályája van?
- Az adott alhéjnak hány darab elektrópályája van?
- Az adott héjnak hány darab elektrónja lehet maximálisan?
- Az adott alhéjnak hány darab elektrónja lehet maximálisan?
- Az adott héj, alhéj esetén a Pauli-féle tilalmi elv szerint milyen mágneses kvantumszámok és mágneses spinquantumszámok lehetségesek?
- Hogyan lehet megadni egy elem atomjának az elektrónkonfigurációját?

Vissza a program elejére      Vissza      Kilépés      Teljes program

```
Teljes program
fokvantumszám(1). fokvantumszám(2). fokvantumszám(3).
fokvantumszám(4). fokvantumszám(5). fokvantumszám(6).
fokvantumszám(7).

mellekkvantumszám(0). mellekkvantumszám(1).
mellekkvantumszám(2). mellekkvantumszám(3).
mellekkvantumszám(4). mellekkvantumszám(5).
mellekkvantumszám(6).

/*mellekkvantumszám_2(L):-fokvantumszám(N), L is N-1.*/

magnesek(-6). magnesek(-5). magnesek(-4).
magnesek(-3). magnesek(-2). magnesek(-1).
magnesek(0). magnesek(1). magnesek(2).
magnesek(3). magnesek(4). magnesek(5). magnesek(6).

/*magnesek_2(M):-mellekkvantumszám_2(M), M>0.
magnesek_2(M):-mellekkvantumszám_2(N), M is -N.*/

spin(0.5). spin(-0.5).

héj(N):- fokvantumszám(N).

alhéj(N,L):- fokvantumszám(N), mellekkvantumszám(L),L<N.

héj_pályák(N,Db):- fokvantumszám(N), Db is N*N.

héj_elektronjai(N,Edb):- fokvantumszám(N), Edb is 2*N*N.
```

## C függelék – Gyakorló feladatsorok

### Gyakorló feladatsor I.

1. Írjon programot, amely kiírja a képernyőre a „Hello World” szöveget!
2. Írjon programot, amely kiszámítja és kiírja egy téglalap területét és kerületét, ha megadjuk a két oldalának a hosszát!
3. Írjon programot, amely adott oldalakból kiszámítja a téglatest felszínét és térfogatát, ha megadjuk a két oldalának a hosszát!
4. Hogyan számítható ki egy kör területe, kerülete a sugarából?
5. Írjon programot, amely 3 beolvasott számot összeszoroz!
6. Írja ki 3 beolvasott szám számtani közepét!
7. Írjon olyan programot, amelyik szögmértékeket vált át fokról radiánra és viszont (radiánról fokra)!
8. Írjon olyan programot, amelyik Ft-ban megadott összeget vált át svájci frank-ra!
9. Írjon olyan programot, amelyik kiszámolja egy autó fogyasztását ismerve a megtett kilométert, valamint az elfogyasztott üzemanyagot!
10. Írjon programot, amely kiszámítja és kiírja egy négyzet területét és kerületét!
11. Írjon programot, amely kiszámítja és kiírja egy kocka felszínét és térfogatát!
12. Írjon olyan programot, amelyik km-ben megadott távolságot vált át mérföldre és fordítva mérföldről km-be!
13. Írjon olyan programot, amelyik inch-ben megadott távolságot vált át méterre, deciméterre, centiméterre!
14. Írjon olyan programot, amelyik láb-ban megadott távolságot vált át méterre, deciméterre, centiméterre!
15. Írjon olyan programot, amelyik °C-ban megadott hőmérsékletet vált át Fahrenheit-re!
16. Írjon olyan programot, amelyik kiszámolja, hogy egy megabájtban megadott adatmennyiség mennyi kilobájtban, illetve bájtban kifejezve.
17. Írjon olyan programot, amelyik kiszámolja, hogy egy kilogrammban megadott tömeg hány dekagrammnak, illetve hány grammnak felel meg.
18. Írjon olyan programot, amelyik kiszámolja, hogy egy literben megadott űrtartalom hány  $\text{m}^3$ -nek,  $\text{dm}^3$ -nek,  $\text{cm}^3$ -nek felel meg.
19. Írjon olyan programot, amelyik kiszámolja, hogy egy órában megadott időtartam hány percnak, másodpercnak felel meg.

## Gyakorló feladatsor II.

### 1. Autók feladat

Ebben a feladatban autót szeretne vásárolni Jutka és Robi. Vannak autók, amelyek mindkettőjüknek tetszenek, vannak azonban olyanok is, amelyek csak az egyiknek. Emellett olyan autók is a kedvenceik közé tartoznak, amelyre sajnos nincs pénzük.

Az autókat halmazokba csoportosíthatjuk a megadott feltételek szerint (melyik, kinek tetszik, van-e rá pénz). Ha egy autót több halmazba is be kellene sorolni, akkor értelemszerűen a halmazok metszeteibe helyezzük el.

Kérem tekintse meg a feladat illusztrálására szolgáló animációt (*Autók\_feladat.swf*)!

- 1.1. Tetszik-e Robinak a Skoda Fabia? A kód megírásához és a célállítás megfogalmazásához tekintse meg az *Autók\_1\_1.swf* animációt!
- 1.2. Tetszik-e Robinak a Wartburg? A kód megírásához és a célállítás megfogalmazásához tekintse meg az *Autók\_1\_2.swf* animációt!
- 1.3. Milyen autók tetszenek Robinak? A kód megírásához és a célállítás megfogalmazásához tekintse meg az *Autók\_1\_3.swf* animációt!
- 1.4. Milyen autót vegyenek, ha abban egyeztek meg, hogy az autó mind a kettőjüknek kedves legyen, valamint csak olyat választanak, amelyre van pénzük. A kód megírásához és a célállítás megfogalmazásához tekintse meg az *Autók\_1\_4.swf* animációt!

Ezek után önállóan oldja meg a további feladatokat:

- 1.5. Melyik autó tetszik Jutkának is és Robinak is?
- 1.6. Melyik autóra van pénzük, amelyik Jutkának tetszik?
- 1.7. Melyek azok az autók, amelyekre nincs pénzük, de tetszenek mindkettőjüknek.
- 1.8. Melyek azok az autók, amelyekre ugyan van pénzük, de egyikőjüknek sem tetszenek?

## 2. Kedvenc állatok feladat

Ebben a feladatban megadjuk, hogy ki milyen házi állatot/kisállatot tartott már.

**Robi:** kutya, macska, aranyhal

**Réka:** teknős, kutya, macska, aranyhal

**Nóra:** kutya, macska, aranyhal, tengerimalac

**Jutka:** kutya, macska, papagáj, pinty, tengerimalac

**Ádám:** teknős, kutya, macska, aranyhal, papagáj, pinty, tengerimalac

- 2.1. Milyen kis kedvencet tartott már mindenki? A kód megírásához és a célállítás megfogalmazásához tekintse meg a *Kedvenc\_állatok\_feladat.swf* animációt!

Ezek után önállóan oldja meg a további feladatokat:

- 2.2. Tartott-e már Réka tengerimalacot?
- 2.3. Tartott-e már Jutka papagájt?
- 2.4. Kik azok, akik már tartottak a pintyet?
- 2.5. Milyen állatai voltak már Nórának?
- 2.6. Melyek azok az állatok, amelyeket Réka is és Nóra is tartott?

## 3. Játékok feladat

Ebben a feladatban megadjuk, hogy ki mit szeret játszani.

**Jutka:** labdázni, tollasozni, ugrókötelezni

**Réka:** gombfocizni, labdázni, kártyázni, sakkozni, társasjátékozni, tollasozni, ugrókötelezni

**Nóra:** labdázni, társasjátékozni, ugrókötelezni

**Robi:** gombfocizni, labdázni, kártyázni, sakkozni, társasjátékozni, tollasozni

- 3.1. Szeret-e Réka sakkozni? A kód megírásához és a célállítás megfogalmazásához tekintse meg a *Játékok\_feladat.swf* animációt!

Ezek után önállóan oldja meg a további feladatokat:

- 3.2. Szeret-e Jutka gombfocizni?
- 3.3. Mivel szeret Nóra játszani?
- 3.4. Kik szeretnek társasjátékozni?
- 3.5. Mit szeretnek mindannyian játszani?
- 3.6. Kik szeretnek labdázni és kártyázni?

## Gyakorló feladatsor III.

### 1. Rákóczi családfa feladat

Ebben a feladatban a Rákóczi családfa egy részletét fogjuk tanulmányozni. A feladatban megadjuk az egyes személyek születési és halálozási évét, ki kinek volt az édesanyja, valamint ki kinek volt a házastársa.

#### személy

név	születési év	halálozási év
Zrínyi Ilona	1643	1703
Zrínyi Péter	1621	1671
Frangepán Katalin	1625	1673
Rákóczi Julianna	1672	1717
II. Rákóczi Ferenc	1676	1735
I. Rákóczi Ferenc	1645	1676
II. Rákóczi György	1621	1660
Báthory Zsófia	1629	1680
Sarolta Amália	1679	1722
Rákóczi Lipót	1696	1699
Rákóczi József	1700	1738
Rákóczi György	1701	1756

#### anya

anya	gyermek
Zrínyi Ilona	Rákóczi Julianna
Zrínyi Ilona	II. Rákóczi Ferenc
Sarolta Amália	Rákóczi Lipót
Sarolta Amália	Rákóczi József
Sarolta Amália	Rákóczi György
Báthory Zsófia	I. Rákóczi Ferenc
Frangepán Katalin	Zrínyi Ilona

#### házastárs

férj	feleség
Zrínyi Péter	Frangepán Katalin
I. Rákóczi Ferenc	Zrínyi Ilona
II. Rákóczi György	Báthory Zsófia
II. Rákóczi Ferenc	Sarolta Amália

Kérem tekintse meg a feladat illusztrálására szolgáló animációt (*Rákóczi\_családfa\_feladat.swf*)!

- 1.1. Ki volt az édesanyja II. Rákóczi Ferencnek? A kód megírásához és a célállítás megfogalmazásához tekintse meg az *Rákóczi\_családfa\_1\_1.swf* animációt!
- 1.2. Ki volt a házastársa Báthory Zsófiának?  
Segítség: *Rákóczi\_családfa\_1\_2.swf* animációt!
- 1.3. Igaz-e, hogy Zrínyi Péter apja II. Rákóczi Ferencnek?  
Segítség: *Rákóczi\_családfa\_1\_3.swf* animációt!
- 1.4. Igaz-e, hogy Zrínyi Péter apja Zrínyi Ilonának?  
Segítség: *Rákóczi\_családfa\_1\_4.swf* animációt!

Ezek után önállóan oldja meg a további feladatokat:

- 1.5. Ki volt a házastársa II. Rákóczi Ferencnek?
- 1.6. Ki volt az édesanyja Rákóczi Juliannának?
- 1.7. Ki volt az édesapja Rákóczi Juliannának?
- 1.8. Ki volt az apai nagyapja II. Rákóczi Ferencnek?
- 1.9. Ki volt az anyai nagyapja II. Rákóczi Ferencnek?



## 2. Európai országok feladat

Ebben a feladatban a európai országok egy csoportját fogjuk tanulmányozni. A feladatban megadjuk az országok, valamint a régiók adatait a következő formában:

ország('Magyarország', 'Budapest', 10053000, 93036, 1014, 10370, 2004, 1).  
ország('Szlovákia', 'Pozsony', 5401000, 49037, 2655, 8775, 2004, 1).  
ország('Csehország', 'Prága', 10325941, 78866, 1602, 13848, 2004, 1).  
ország('Románia', 'Bukarest', 22303552, 238391, 2543, 7311, 2007, 2).  
ország('Ukrajna', 'Kijev', 47425336, 603700, 2061, 1020, 0, 2).  
ország('Szerbia', 'Belgrád', 9400000, 88361, 2755, 3200, 0, 3).  
ország('Horvátország', 'Zágráb', 4493312, 56542, 1831, 4550, 0, 3).  
ország('Olaszország', 'Róma', 58883958, 301318, 4807, 30200, 1957, 3).  
ország('Spanyolország', 'Madrid', 45116894, 504782, 3718, 27767, 1986, 3).  
ország('Franciaország', 'Párizs', 63587700, 674843, 4807, 29316, 1957, 4).  
ország('Lengyelország', 'Varsó', 38626349, 312685, 2499, 11000, 2004, 1).  
ország('Hollandia', 'Amszterdam', 16491461, 41526, 322, 29412, 1957, 4).  
ország('Finnország', 'Helsinki', 5282301, 338145, 1328, 37504, 1995, 5).  
ország('Dánia', 'Koppenhága', 5447084, 43094, 173, 47984, 1973, 5).  
ország('Svédország', 'Stockholm', 9122269, 449964, 2111, 42694, 1995, 5).

régió(1, 'Közép-Európa').

régió(2, 'Kelet-Európa').

régió(3, 'Dél-Európa').

régió(4, 'Nyugat-Európa').

régió(5, 'Észak-Európa').

### ország argumentumai:

- országnév, főváros, lakosság, terület, legmagasabb pont, egy főre eső GDP,
- EU csatlakozás évszáma, régiókód

### régió argumentumai:

- régiókód, régiónév.

### Feladatok:

- 2.1. Írassuk ki egy megadott ország fővárosát.
- 2.2. Írassuk ki egy megadott régió országait.
- 2.3. Adjuk meg a népes országokat (népesség > 15 millió).
- 2.4. Adjuk meg a nem népes országokat (népesség < 6 millió).
- 2.5. Írassuk ki az országok népsűrűségét.
- 2.6. Listázzuk ki a gazdag országokat (GDP/fő>15000).
- 2.7. Listázzuk ki a szegény országokat (GDP/fő<5000).
- 2.8. Adjuk meg a nagy országokat (terület>150000).
- 2.9. Adjuk meg a kis országokat (terület<50000).
- 2.10. Írassuk ki az alföldi országokat (legmagasabb pont < 500).

## Gyakorló feladatsor IV.

### 1. Ládák 1

Shakespeare *Velencei kalmár*-jában Portiának volt három ládikája – egy arany, egy ezüst és egy ólom –, amelyek egyikében Portia képe rejtőzött. Kérőjének választania kellett egyet a ládikák közül, és ha elég szerencsés (vagy elég bölcs) volt ahhoz, hogy a képet tartalmazó ládikát válassza, akkor igényt tarthatott Portia kezére. A ládikákon lévő egy-egy felirat segítette a kérőt a bölcs választásban.

A kép ebben a  
ládikában van.

**Arany**

A kép nem  
ebben a  
ládikában van.

**Ezüst**

A kép nem az  
aranyládikában  
van.

**Ólom**

Portia annyit mondott kérőjének, hogy a három állítás közül legfeljebb egy igaz. Melyik ládikát válassza a kérő?

### 2. Ládák 2

Alább látható 3 ládikó. A 3 ládikóra írt egy-egy állítás között van igaz és van hamis is.

Az arany nem a  
2.ládikóban van.

**1.ládikó**

Az arany nem  
ebben a  
ládikóban van.

**2.ládikó**

Itt van az arany.

**3.ládikó**

Melyikben van az arany?

### 3. Szerények

Négy szerénység – Szeréna, Szergej, Szervác, Szeráf – szerényen a következőket mondták.

Szeréna: „Szergej a legszerényebb.”

Szergej: „Szervác a legszerényebb.”

Szervác: „Nem én vagyok a legszerényebb.”

Szeráf: „Nem én vagyok a legszerényebb.”

A négy állítás közül csak egy volt igaz. A négy közül ki a legszerényebb?

## Gyakorló feladatsor V.

### 1. Mappák feladat

Ebben a feladatban egy mapparendszer lesz a vizsgálatunk tárgya. Kérem tekintse meg a feladat illusztrálására szolgáló animációt (*Mappák\_feladat.swf*)!

- 1.1. Benne van-e a **feladatok** mappa az **alapismeretek** mappában? A kód megírásához és a célállítás megfogalmazásához tekintse meg az *Mappák\_1\_1.swf* animációt!
- 1.2. Benne van-e a **gyakorlat** mappa az **alapismeretek** mappában? A kód megírásához és a célállítás megfogalmazásához tekintse meg az *Mappák\_1\_2.swf* animációt!

Ezek után önállóan oldja meg a további feladatokat:

- 1.3. Benne van-e az **alapismeretek** mappa a **gyakorlat** mappában?
- 1.4. Mely mappák vannak az **excel** mappában?
- 1.5. Mely mappák vannak a **feladatok** mappában?
- 1.6. Benne van-e a **cellahivatkozások** mappa a **feladatok** mappában?
- 1.7. Benne van-e a **cellahivatkozások** mappa az **elmélet** mappában?

### 2. Árpád-házi uralkodók feladat

Ebben a feladatban az Árpád-ház uralkodóit fogjuk vizsgálni. Kérem tekintse meg a feladat illusztrálására szolgáló animációt (*Árpád\_ház\_feladat.swf*)!

Ezután önállóan oldja meg a további feladatokat:

- 2.1. Fia-e I. István Gézának?
- 2.2. Fia-e Vazul Gézának?
- 2.3. Kik Vazul fiai?
- 2.4. Ki az apja Gézának?
- 2.5. Készítsen egy olyan szabályt, amely megadja egy ember (pl.: II. András) utódainak a nevét. (Használjon rekurziót a szabály megalkotásakor!)
- 2.6. Az előző szabály segítségével adja meg egy ember őseinek a nevét.
- 2.7. Készítse el a nagyapja szabályt, majd adja meg egy uralkodó (pl. I. Géza) nagyapját/unokáját.
- 2.8. Készítse el a testvér szabályt, majd adja meg egy uralkodó (Pl. Szent László) testvérét (testvéreit).
- 2.9. Készítse el az unokatestvér szabályt, majd adja meg egy uralkodó (Pl. Salamon) unokatestvérét (unokatestvéreit).

### 3. Rákóczi családfa feladat (rekurzióval bővítve)

Ebben a feladatban folytatjuk a Rákóczi családfa tanulmányozását.

Kérem tekintse meg ismét a feladat illusztrálására szolgáló animációt (*Rákóczi\_családfa\_feladat.swf*)!

- 3.1. II. Rákóczi Ferenc anyai ágon leszármazottja-e Zrínyi Ilonának? A kód megírásához és a célállítás megfogalmazásához tekintse meg az *Rákóczi\_családfa\_3\_1.swf* animációt!
- 3.2. II. Rákóczi Ferenc anyai ágon leszármazottja-e Frangepán Katalinnak? A kód megírásához és a célállítás megfogalmazásához tekintse meg az *Rákóczi\_családfa\_3\_2.swf* animációt!
- 3.3. Igaz-e, hogy Rákóczi Lipót apai ágon leszármazottja II. Rákóczi Ferencnek? A kód megírásához és a célállítás megfogalmazásához tekintse meg az *Rákóczi\_családfa\_3\_3.swf* animációt!

Ezek után önállóan oldja meg a további feladatokat:

- 3.4. Igaz-e, hogy I. Rákóczi Ferenc apai ágon leszármazottja Zrínyi Péternek?
- 3.5. Igaz-e, hogy I. Rákóczi Ferenc apai ágon leszármazottja II. Rákóczi Györgynek?
- 3.6. Igaz-e, hogy II. Rákóczi Ferenc apai ágon leszármazottja Zrínyi Péternek?
- 3.7. Igaz-e, hogy II. Rákóczi Ferenc apai ágon leszármazottja II. Rákóczi Györgynek?
- 3.8. Készítse el az anyós szabályt, majd adja meg egy személy (pl. Zrínyi Ilona) anyósát!
- 3.9. Hogyan lehet megkapni azt, hogy kinek az anyósa Zrínyi Ilona?
- 3.10. Készítse el az após szabályt, majd adja meg egy személy (pl. Sarolta Amália) apósát!
- 3.11. Hogyan lehet megkapni azt, hogy kinek az apósa Zrínyi Péter?

## Gyakorló feladatsor VI.

### 1. Számok kiírása feladat

Ebben a feladatban bekérünk egy pozitív egész számot,  $s$  addig, 1-től kezdődően kiíratjuk a számokat egyesével. (Pl.: 10-re kiíratjuk, hogy 1, 2, 3, 4, 5, 6, 7, 8, 9, 10.)

Kérem tekintse meg a feladat illusztrálására szolgáló animációt (*Számok\_feladat.swf*)!

Ezután önállóan oldja meg a további feladatokat:

- 1.1. Írassa ki 1-től 50-ig a számokat egyesével.
- 1.2. Írassa ki 2-től kezdődően az első 10 páros számot!
- 1.3. Írassa ki 2-től kezdődően az első 50 páros számot!
- 1.4. Írassa ki a 3-as szorzótáblát!
- 1.5. Írassa ki a 3-as szorzótábla első 50 elemét!

### 2. $n$ faktoriális ( $n!$ ) feladat

Ebben a feladatban az  $n!$  értékét fogjuk meghatározni rekurzió segítségével.

Kérem tekintse meg a feladat illusztrálására szolgáló animációt (*n\_faktoriális\_feladat.swf*)!

- 2.1. Mennyi  $0!$  értéke? A kód megírásához és a célállítás megfogalmazásához tekintse meg az *n\_faktoriális\_2\_1.swf* animációt!
- 2.2. Mennyi  $1!$  értéke? A kód megírásához és a célállítás megfogalmazásához tekintse meg az *n\_faktoriális\_2\_2.swf* animációt!
- 2.3. Mennyi  $2!$  értéke? A kód megírásához és a célállítás megfogalmazásához tekintse meg az *n\_faktoriális\_2\_3.swf* animációt!

Ezek után önállóan oldja meg a további feladatokat:

- 2.4. Mennyi  $5!$  értéke?
- 2.5. Készítsen olyan programot, amelyik összeadja 1-től  $n$ -ig az egész számokat!
- 2.6. Mennyi 1-től 10-ig a pozitív egész számok összege?
- 2.7. Mennyi 1-től 50-ig a pozitív egész számok összege?

### 3. Célbalövés feladat

Ebben a feladatban célbalövünk és különféle vizsgálatokat teszünk a lövések eredményeivel kapcsolatban. Kérem tekintse meg a feladat illusztrálására szolgáló animációt (*Célbalövés\_feladat.swf*)!

- 3.1. Sikerült-e 10-est lőni a sorozatban (bennevan-e a 10-es a listában)? A kód megírásához és a célállítás megfogalmazásához tekintse meg az *Célbalövés\_3\_1.swf* animációt!
- 3.2. Sikerült-e 9-est lőni a sorozatban (bennevan-e a 9-es a listában)? A kód megírásához és a célállítás megfogalmazásához tekintse meg az *Célbalövés\_3\_2.swf* animációt!
- 3.3. Egy listában tároljuk kockadobások eredményeit.  
Pl.: [6, 5, 1, 2, 2, 1, 3, 5, 6, 1, 3, 1]  
Sikerült-e 6-ost dobni?
- 3.4. Egy listában tároljuk kockadobások eredményeit.  
Pl.: [6, 5, 1, 2, 2, 1, 3, 5, 6, 1, 3, 1]  
Sikerült-e 4-est dobni?

### 4. Kártya feladat

Ebben a feladatban egy magyar kártya csomagot fogunk vizsgálni. Kérem tekintse meg a feladat illusztrálására szolgáló animációkat (*Kártya\_feladat.swf*, *Kártya\_feladat\_lap.swf*)!

- 4.1. Benne van-e a piros ász az alábbi kártyakötegben?  
[p7, p8, t7, z10, zf, pal, má, zal, p10, pá, m9, tal]
- 4.2. Benne van-e a makk 7-es az alábbi kártyakötegben?  
[p7, p8, t7, z10, zf, pal, má, zal, p10, pá, m9, tal]
- 4.3. Benne van-e a piros ász az alábbi kártyakötegben?  
[lap(p,7), lap(p,8), lap(t,7), lap(z,10), lap(z,f), lap(p,al), lap(m,á), lap(z,al), lap(p,10), lap(p,á), lap(m,9), lap(t,al)]
- 4.4. Benne van-e a zöld 8-as az alábbi kártyakötegben?  
[lap(p,7), lap(p,8), lap(t,7), lap(z,10), lap(z,f), lap(p,al), lap(m,á), lap(z,al), lap(p,10), lap(p,á), lap(m,9), lap(t,al)]

## D függelék – Bemeneti teszt

### Problémamegoldó feladatok

#### Bemeneti teszt

**Időtartam: 60 perc**

#### Adatok:

Név: .....Osztály: .....

Legutóbbi, év végi Matematika jegye: .....

#### Feladatok:

1. Az Óperencián túl 1 év 3 hónapból áll, s egy hónap 9 napból. Hány napig tart túl az Óperencián a 7 éves szolgálat? **6 pont**

Válasz:

Indoklás:

2. Hány fácán tömegével egyenlő 1 róka tömege, ha 1 nyúl tömege egyenlő 2 fácánéval, és 2 róka tömege egyenlő 5 nyúléval. **6 pont**

Válasz:

Indoklás:

3. **10 pont**

A következő feladatban megfelelő időpontot kell találnod egy mozizáshoz.

Gergő 15 éves. Egy közös mozizást szeretne szervezni az egyhetes tavaszi szünet idején két barátjával, akik vele egyidősek. A tavaszi szünet március 24-én, szombaton kezdődik, és április 1-ig, vasárnapig tart.

Gergő megkérdezte a barátait, hogy mely napok, és mely időpontok lennének alkalmasak nekik a mozizásra. A következő válaszokat kapta:

**Attila:** „Hétfő és szerda délután 2.30 és 3.30 között otthon kell maradnom, és a hangszeremen kell gyakorolnom.”

**Sanyi:** „Nekem vasárnaponként a nagymamámat kell meglátogatnom, úgyhogy a vasárnap nem jöhet szóba. A Pokamint már láttam, és nem akarom még egyszer megnézni.”

Gergő szülei ragaszkodnak ahhoz, hogy fiuk csak korának megfelelő filmet nézzén meg, és ne menjen gyalog haza. Vállalják, hogy hazaszállítják a fiukat bármilyen időpontban, egészen este 10-ig.

Gergő átnézte a szünidő hetére vonatkozó moziműsort, és a következőket találta:

<b>SZIVÁRVÁNY MOZI</b> Jegyelővétel: 429-6877 24 órás telefonszolgálat: 429-6878 Akciós nap keddenként: minden film csak 400 Ft <b>Moziműsor két hétre, március 23., péntektől:</b>			
<b>Kölykök kelepcebén</b> 113 perc 14.00 (csak hétfőtől péntekig) 21.35 (csak szombat-vasárnap)		<b>Pokamin</b> 105 perc 13.40 (minden nap) 16.35 (minden nap)	
Csak 12 éven felülieknek!		Szülői felügyelet javasolt. Minden korosztály számára ajánlott, de néhány jelenet sértheti a kisgyermek érzékenységét.	
<b>Szörnyek a mélyből</b> 164 perc 19.55 (csak pénteken és szombaton)		<b>A rejtély</b> 144 perc 15.00 (csak hétfőtől péntekig) 18.00 (csak szombat-vasárnap)	
Csak 18 éven felülieknek!		Csak 12 éven felülieknek!	
<b>A ragadozó</b> 148 perc 18.30 (minden nap)		<b>A vadon királya</b> 117 perc 14.35 (csak hétfőtől péntekig) 18.50 (csak szombat-vasárnap)	
Csak 18 éven felülieknek!		Minden korosztály számára ajánlott.	

### 1. feladat

Összevetve a filmre vonatkozó információkat és azokat, amelyeket Gergő a barátaitól kapott, melyek azok a filmek a hatból, amelyeket Gergő és barátai megnézhetnek?

Válaszodat az „Igen” vagy a „Nem” bekarikázásával jelöld minden filmcím esetén!

Filmek	A három fiú megnézheti az adott filmet?
Kölykök kelepcebén	Igen / Nem
Szörnyek a mélyből	Igen / Nem
A ragadozó	Igen / Nem
Pokamin	Igen / Nem
A rejtély	Igen / Nem
A vadon királya	Igen / Nem

### 2. feladat

Ha a három fiú úgy dönt, hogy a „Kölykök kelepcebén” című filmet nézik meg, akkor az alábbi időpontok közül melyik megfelelő a számukra?

- A Március 26., hétfő.
- B Március 28., szerda.
- C Március 30., péntek.
- D Március 31., szombat.
- E Április 1., vasárnap.



4. Hárman, Anikó, Balázs és Cili beszélgettek.

7 pont

Anikó azt mondja: „Balázs hazudik.”

Balázs azt mondja: „Cili hazudik.”

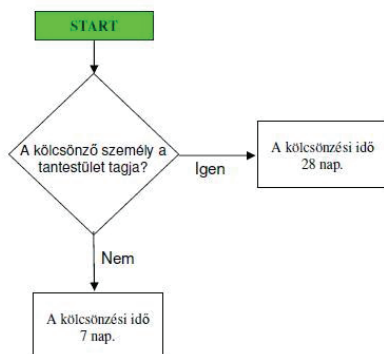
Cili azt mondja: „Anikó és Balázs hazudik.”

Ki mond igazat, ki hazudik?

- 5.

10 pont

Az **Erőss János Gimnázium** könyvtárában a következő egyszerű könyvkölcsönzési rendszer működik: A tantestületi tagok részére 28 nap és a diákok részére 7 nap a kölcsönzési idő. A következő döntési folyamatára szemlélteti ezt a rendszert:



A Zölderdei Középiskola könyvtára egy hasonló, de valamivel bonyolultabb kölcsönzési rendszert használ.

- Minden „Védett” besorolású kiadvány kölcsönzési ideje 2 nap.
- A „Védett” listán **nem** szereplő könyvek kölcsönzési ideje (nem számítva ide a folyóiratokat) a tantestületi tagok részére 28 nap, a diákok részére 14 nap.
- A „Védett” listán **nem** szereplő folyóiratok kölcsönzési ideje mindenki számára 7 nap.
- Semmit sem kölcsönözhetnek ki azok, akiknél lejárt kölcsönzési idejű kiadvány található.

#### 1. feladat

Te a Zölderdei Középiskolába jársz, és nincs nálad lejárt kölcsönzési idejű kiadvány. Egy, a „Védett” listán **nem** szereplő könyvet szeretnél kikölcsönözni. Mennyi időre kölcsönözheted ki a könyvet?

Válasz: ..... napra.

## 2. feladat

Készíts egy olyan döntési folyamatábrát, amely alapján meg lehet tervezni egy automatikus könyv- és folyóirat-kölcsönzési rendszert a Zölderdei Középiskola könyvtára számára. A rendszer a lehető leghatékonyabb legyen (azaz minél kevesebb lépésből álljon)! Fontos, hogy minden lépésnek csak két kimenete lehet, és a kimeneteket megfelelően kell elnevezni (pl. „Igen” és „Nem”).



6. Anna, Béla és Cili szétosztanak 9 golyót úgy, hogy mindegyikük különböző számú golyót kap, és mindenki kap legalább egyet. Az őket kérdező Dénesnek a következőket mondják:

Anna: „Nálam legfeljebb 5 golyó van.”

**8 pont**

Béla: „Annánál van a legtöbb golyó.”

Cili: „Béla nem hazudik.”

Kinél van a legtöbb golyó és mennyi, ha tudjuk, hogy hármójuk közül pontosan egy gyerek hazudott?

*Válasz:*

*Indoklás:*

7. Egy öttagú társaságban, ahol 3 lány és 2 fiú van, mindenki mindenkire rámutatott, és kimondta a következő két mondat valamelyikét: „Ő fiú.” vagy „Ő lány.”
- Mindegyik kimondott mondat igaz volt.

**9 pont**

Hány mondat hangzott el összesen?

*Válasz:*

*Indoklás:*

Döntse el, hogy az alábbi állítások igaz vagy hamis értékűek?

- Mind a két mondat ugyanannyiszor hangzott el.
- Az „Ő lány.” mondat néggyel többször hangzott el, mint az „Ő fiú.”
- Az „Ő lány.” mondat nyolcszor hangzott el.
- Az „Ő fiú.” mondat hatszor hangzott el.

8. Portiának volt három ládikája – egy arany, egy ezüst és egy ólom –, amelyek egyikében Portia képe rejtőzött. Kérőjének választania kellett egyet a ládikák közül, és ha elég bölcs volt ahhoz, hogy a képet tartalmazó ládikát válassza, akkor igényt tarthatott Portia kezére. A ládikákon lévő egy-egy felirat segítette a kérőt a bölcs választásban. **8 pont**

A kép ebben a ládikában van.	A kép nem ebben a ládikában van.	A kép nem az aranyládikában van.
Arany	Ezüst	Ólom

Portia annyit mondott kérőjének, hogy a három állítás közül legfeljebb egy igaz. Melyik ládikát válassza a kérő?

*Válasz:*

*Indoklás:*

9. Egy távoli bolygón anka nyelven beszélnek. Négy mondat fordítását megadjuk ankául: **8 pont**
- Leesett a kifli. = Ham bam.
  - Szeretem a levest. = Vele memme.
  - Leesett a fáról. = Bam fam.
  - Szeretem a kakaót. = Dudu memme.

Hogyan mondhatják ankául a következő mondatot, ha az anka nyelvben minden szónak egy jelentése van?

**Szeretem összeszedni a fáról leesett almákat.**

- a. Memme venne fam bam ma.
- b. Memme ham fam bam ma.
- c. Vele dudu fam bam ham.
- d. Memme venne ham bam ma.
- e. Dudu venne bam fam al.

*Válasz:*

*Indoklás:*

10. Az iskolai futóverseny döntőjébe 5-en jutottak: Aladár, Béla, Csaba, Dezső és Endre. A verseny után az eredményről a következőt mondták: **10 pont**

Aladár: - Dezső második lett. Engem csak ketten előztek meg.  
Béla: - Én győztem. A második Csaba lett.  
Csaba: - Harmadik lettem. Bélát mindenki megelőzte.  
Dezső: - Második helyen végeztem. Endre negyedik lett.  
Endre: - Egyetlen futót előztem meg. A versenyt Aladár nyerte.

Ki nyerte a versenyt, ha tudjuk, hogy mindegyik versenyzőnek az egyik állítása igaz, a másik hamis és nem volt holtverseny?

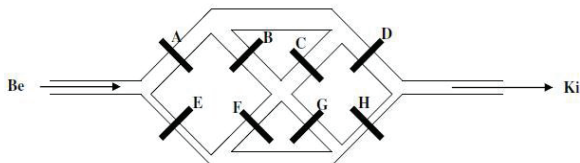
11. Négy szerénység – Szeréna, Szergej, Szervác, Szeráf – szerényen a következőket mondták. **8 pont**

Szeréna: „Szergej a legszerényebb.”  
Szergej: „Szervác a legszerényebb.”  
Szervác: „Nem én vagyok a legszerényebb.”  
Szeráf: „Nem én vagyok a legszerényebb.”  
A négy állítás közül – mint utóbb szerényen kiderült – csak egy volt igaz. A négy szerénység közül ki a legszerényebb?

*Válasz:*

*Indoklás:*

12. Az alábbi ábrán egy öntözőrendszer látható, amellyel különböző növényeket öntöznek. Az A-tól H-ig elnevezett zsilipkapukat kell kinyitni, illetve bezárni ahhoz, hogy odaeresszék a vizet, ahová szükséges. Amikor egy zsilipkapu zárva van, nem folyik át rajta a víz. **10 pont**

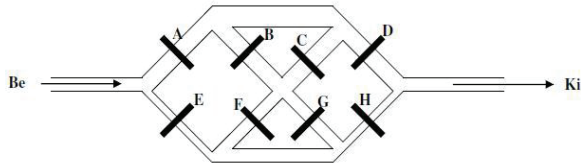


Balázs észrevette, hogy a víz nem mindig arra folyik, amerre kellene. Arra gondol, hogy az egyik zsilipkapu beragadt, azaz akkor sem nyílik ki, ha a kapcsoló “Nyitva” állásban van. Balázs az alábbi táblázat szerint állítja be a zsilipkapukat azért, hogy leellenőrizze a működésüket.

**1.táblázat**

A	B	C	D	E	F	G	H
Nyitva	Zárva	Nyitva	Nyitva	Zárva	Nyitva	Zárva	Nyitva

Az 1.táblázatban megadott zsilipkapu-beállítások alapján rajzolja be az alábbi ábrába a vízfolyás összes lehetséges útját! Feltételezzük, hogy a zsilipkapuk működnek a beállításoknak megfelelően.



Balázs észrevette, hogy ha az 1.táblázat szerint állítja be a zsilipkapukat, akkor nem folyik át a víz a rendszeren. Ezek szerint legalább egy “Nyitva” állásban lévő zsilipkapu beragadt. Döntse el, hogy az alábbi problémák előfordulása esetén végigfolyik-e a víz a rendszeren! Válaszát minden esetben az „Igen” vagy a „Nem” bekarikázásával jelölje!

Probléma	Átfolyik a víz az egész rendszeren?
Az A zsilipkapu beragadt. Minden más zsilipkapu az 1. táblázat szerint működik.	Igen / Nem
A D zsilipkapu beragadt. Minden más zsilipkapu az 1. táblázat szerint működik.	Igen / Nem
Az F zsilipkapu beragadt. Minden más zsilipkapu az 1. táblázat szerint működik.	Igen / Nem

Balázs azt szeretné leellenőrizni, hogy beragadt-e a D zsilipkapu. A 2.táblázatba írjon olyan beállításokat, amelyekkel bizonyítható, hogy a D zsilipkapu beragadt, holott “Nyitva” állásba kapcsolták!

Zsilipkapu-beállítások (vagy “Nyitva” vagy ”Zárva”):

**2.táblázat**

A	B	C	D	E	F	G	H

## E függelék – A bemeneti tesztek eredményei

### József Attila Szakközépiskola, Szakiskola és Kollégium

#### Bemeneti teszt - kontroll csoport

Ssz.	Név	Csop.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	Össz.
1.	Tanuló 1	0	6	0	4	2	3	0	2	0	2	0	0	0	19
2.	Tanuló 2	0	6	0	8	0	0	3	3	0	4	0	0	0	24
3.	Tanuló 3	0	6	2	10	0	2	1	2	0	0	0	0	0	23
4.	Tanuló 4	0	0	0	9	3	2	0	3	0	2	0	1	0	20
5.	Tanuló 5	0	2	2	0	3	0	3	3	0	0	1	1	0	15
6.	Tanuló 6	0	6	6	10	2	2	2	1	0	8	1	3	10	51
7.	Tanuló 7	0	6	2	10	0	6	2	3	2	6	1	1	5	44
8.	Tanuló 8	0	6	2	10	0	3	2	1	0	0	1	1	2	28
9.	Tanuló 9	0	6	2	10	1	3	1	3	0	1	2	1	1	31
10.	Tanuló 10	0	0	0	8	3	2	2	1	0	0	2	0	6	24
11.	Tanuló 11	0	6	5	8	3	0	2	3	4	6	0	0	0	37
12.	Tanuló 12	0	6	6	9	1	0	0	5	0	0	0	1	0	28
13.	Tanuló 13	0	6	0	8	0	2	0	0	0	3	1	0	6	26
14.	Tanuló 14	0	0	6	8	0	2	0	3	0	2	0	3	4	28
15.	Tanuló 15	0	6	6	9	0	2	0	4	0	0	0	0	0	27
16.	Tanuló 16	0	6	0	0	3	0	1	3	0	2	0	3	0	18
17.	Tanuló 17	0	6	2	8	4	2	3	3	2	0	0	1	3	34
18.	Tanuló 18	0	6	6	9	2	0	2	2	0	8	0	1	2	38
19.	Tanuló 19	0	6	3	8	2	2	1	8	0	2	0	0	8	40
20.	Tanuló 20	0	0	6	10	0	0	0	0	0	4	0	0	0	20
21.	Tanuló 21	0	6	3	8	2	0	1	0	2	8	1	1	2	34
22.	Tanuló 22	0	6	6	10	0	3	8	3	0	0	0	0	0	36
23.	Tanuló 23	0	6	0	10	2	1	0	3	2	0	0	0	0	24
24.	Tanuló 24	0	6	2	10	1	0	1	3	0	0	0	1	3	27
25.	Tanuló 25	0	6	0	2	0	2	0	2	0	0	0	0	0	12
26.	Tanuló 26	0	2	2	7	0	0	0	4	0	0	0	0	0	15
27.	Tanuló 27	0	0	2	5	2	2	1	3	0	4	2	2	1	24
28.	Tanuló 28	0	0	0	7	0	2	0	0	0	0	0	0	0	9
29.	Tanuló 29	0	6	6	8	1	0	1	7	3	1	0	0	0	33
30.	Tanuló 30	0	6	0	9	0	0	0	7	0	5	0	1	2	30
31.	Tanuló 31	0	6	0	9	0	2	0	1	0	8	0	1	0	27
32.	Tanuló 32	0	0	2	6	3	2	0	3	0	0	0	1	2	19
33.	Tanuló 33	0	6	5	0	0	4	2	1	0	2	1	1	0	22
34.	Tanuló 34	0	6	6	10	0	7	5	8	0	8	8	6	5	69
35.	Tanuló 35	0	6	0	2	3	2	2	6	0	2	1	4	1	29
36.	Tanuló 36	0	6	1	10	0	2	0	0	0	6	0	0	1	26
37.	Tanuló 37	0	6	6	8	1	3	3	8	2	2	4	1	10	54
38.	Tanuló 38	0	6	6	9	0	3	0	2	0	2	0	1	2	31

**József Attila Szakközépiskola, Szakiskola és Kollégium**  
**Bemeneti teszt - kísérleti csoport**

Ssz.	Név	Csop.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	Össz.
1.	Tanuló 1	1	6	6	10	0	0	1	2	0	0	0	0	0	25
2.	Tanuló 2	1	6	6	8	0	2	2	0	0	0	0	0	0	24
3.	Tanuló 3	1	6	0	9	0	2	1	2	0	8	1	0	0	29
4.	Tanuló 4	1	6	6	8	0	3	0	3	0	8	0	0	0	34
5.	Tanuló 5	1	0	3	10	1	2	2	0	0	8	1	1	10	38
6.	Tanuló 6	1	6	0	4	3	0	1	2	0	0	0	0	1	17
7.	Tanuló 7	1	6	2	0	1	0	0	4	0	4	1	0	2	20
8.	Tanuló 8	1	0	2	8	2	4	3	2	0	2	1	3	0	27
9.	Tanuló 9	1	6	6	10	0	0	2	0	0	0	0	0	0	24
10.	Tanuló 10	1	6	6	10	0	0	2	5	0	0	0	1	1	31
11.	Tanuló 11	1	6	6	9	0	10	5	0	0	0	0	0	3	39
12.	Tanuló 12	1	6	2	10	0	2	2	2	0	2	0	0	2	28
13.	Tanuló 13	1	6	6	7	0	10	0	0	0	6	0	0	2	37
14.	Tanuló 14	1	0	2	7	2	2	0	2	0	0	0	2	10	27
15.	Tanuló 15	1	0	2	9	2	0	0	2	0	3	0	2	8	28
16.	Tanuló 16	1	0	2	10	2	3	0	1	0	0	0	0	0	18
17.	Tanuló 17	1	6	6	0	2	10	7	4	5	1	0	8	10	59
18.	Tanuló 18	1	6	6	9	2	3	3	9	4	8	7	1	0	58
19.	Tanuló 19	1	6	4	8	3	2	4	4	0	3	1	3	8	46
20.	Tanuló 20	1	6	0	2	0	0	0	3	0	0	0	0	0	11
21.	Tanuló 21	1	6	2	7	1	5	3	8	0	5	2	1	10	50
22.	Tanuló 22	1	0	4	10	1	0	0	5	0	1	1	1	5	28
23.	Tanuló 23	1	0	3	6	0	3	0	1	0	0	1	1	10	25

# Vak Bottyán János Katolikus Műszaki és Közgazdasági Középiskola

## Bemeneti teszt - kontroll csoport

Sz.	Név	Csop.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	Össz.
1.	Tanuló 1	0	6	6	9	1	10	6	9	0	2	0	1	10	60
2.	Tanuló 2	0	6	2	10	1	2	1	5	0	2	2	1	6	38
3.	Tanuló 3	0	6	6	9	0	7	1	5	0	8	2	1	0	45
4.	Tanuló 4	0	6	6	7	0	10	3	6	0	8	0	0	0	46
5.	Tanuló 5	0	6	6	10	1	2	2	1	0	5	2	3	4	42
6.	Tanuló 6	0	6	6	10	3	10	6	1	4	2	8	0	5	61
7.	Tanuló 7	0	6	6	9	3	8	3	5	4	7	0	0	10	61
8.	Tanuló 8	0	6	2	9	1	3	0	9	0	6	1	0	2	39
9.	Tanuló 9	0	6	6	9	0	10	3	2	0	1	2	1	10	50
10.	Tanuló 10	0	6	6	10	2	10	6	9	7	8	8	6	10	88
11.	Tanuló 11	0	6	6	9	1	10	3	9	0	8	7	1	9	69
12.	Tanuló 12	0	6	0	10	0	8	2	2	0	8	2	4	0	42
13.	Tanuló 13	0	6	6	10	0	10	0	7	0	2	3	3	8	55
14.	Tanuló 14	0	2	6	10	1	6	0	0	0	0	0	0	0	25
15.	Tanuló 15	0	6	6	8	2	4	3	7	0	0	1	0	9	46
16.	Tanuló 16	0	6	6	7	0	10	0	1	0	0	0	0	2	32
17.	Tanuló 17	0	6	0	10	0	2	3	2	0	0	5	0	5	33
18.	Tanuló 18	0	6	4	9	3	5	4	7	4	4	7	1	4	58
19.	Tanuló 19	0	6	0	6	0	2	2	8	0	0	1	0	2	27
20.	Tanuló 20	0	6	6	10	1	6	3	5	0	6	1	1	7	52
21.	Tanuló 21	0	6	6	10	1	10	6	7	2	2	0	1	4	55
22.	Tanuló 22	0	6	6	9	0	8	6	0	0	8	1	1	0	45
23.	Tanuló 23	0	6	2	9	3	7	0	3	0	0	0	0	5	35
24.	Tanuló 24	0	6	6	10	0	8	0	6	2	2	1	1	10	52
25.	Tanuló 25	0	5	4	10	1	5	6	3	2	8	3	0	10	57
26.	Tanuló 26	0	6	3	10	3	3	0	3	0	2	1	0	9	40
27.	Tanuló 27	0	6	6	8	1	8	1	1	0	8	2	1	9	51
28.	Tanuló 28	0	6	0	7	2	2	0	2	4	6	2	1	0	32
29.	Tanuló 29	0	6	6	10	0	6	3	5	0	3	0	1	5	45
30.	Tanuló 30	0	6	6	10	0	4	0	1	0	6	2	0	7	42
31.	Tanuló 31	0	6	6	10	1	6	3	8	0	5	3	1	8	57
32.	Tanuló 32	0	6	6	9	0	4	3	9	0	1	1	1	8	48
33.	Tanuló 33	0	6	4	10	0	6	0	8	2	3	2	2	5	48
34.	Tanuló 34	0	6	6	9	2	2	2	9	0	0	2	1	9	48
35.	Tanuló 35	0	6	6	10	0	2	2	9	0	8	1	1	8	53



**Vak Bottyán János Katolikus Műszaki és Közgazdasági Középiskola**  
**Bemeneti teszt - kísérleti csoport**

Ssz.	Név	Csop.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	Össz.
1.	Tanuló 1	1	6	6	10	0	10	6	7	0	8	3	4	10	70
2.	Tanuló 2	1	6	6	9	0	0	3	6	0	8	1	3	6	48
3.	Tanuló 3	1	6	6	9	2	10	7	8	8	8	2	0	0	66
4.	Tanuló 4	1	6	6	10	0	10	4	9	5	8	2	1	10	71
5.	Tanuló 5	1	6	6	8	1	10	0	9	0	5	5	0	10	60
6.	Tanuló 6	1	6	6	10	1	8	3	1	0	8	3	1	9	56
7.	Tanuló 7	1	0	0	9	1	2	1	4	2	8	2	1	6	36
8.	Tanuló 8	1	6	6	9	1	7	5	6	2	8	2	4	7	63
9.	Tanuló 9	1	6	0	9	2	4	0	5	0	4	2	0	2	34
10.	Tanuló 10	1	6	6	10	5	8	2	8	7	2	4	1	10	69
11.	Tanuló 11	1	6	6	0	7	8	7	4	4	8	2	2	6	60
12.	Tanuló 12	1	6	6	9	3	10	0	9	0	0	3	1	2	49
13.	Tanuló 13	1	0	6	10	4	10	0	9	0	8	7	0	10	64
14.	Tanuló 14	1	2	6	9	1	10	2	8	0	0	2	1	10	51
15.	Tanuló 15	1	6	3	8	2	10	0	1	0	8	1	1	10	50
16.	Tanuló 16	1	6	6	10	2	10	5	9	4	7	0	1	0	60
17.	Tanuló 17	1	6	6	10	2	10	6	9	4	6	7	1	0	67
18.	Tanuló 18	1	6	3	10	2	8	1	4	0	2	1	0	8	45
19.	Tanuló 19	1	6	3	10	0	4	3	1	2	0	2	1	7	39
20.	Tanuló 20	1	6	2	10	1	2	1	2	4	2	1	1	0	32
21.	Tanuló 21	1	0	6	8	1	0	0	6	0	8	0	1	10	40
22.	Tanuló 22	1	6	6	10	2	10	1	3	0	8	3	1	10	60
23.	Tanuló 23	1	0	0	6	2	0	2	4	2	0	1	1	3	21
24.	Tanuló 24	1	6	4	10	0	6	0	7	0	8	1	1	5	48
25.	Tanuló 25	1	6	6	10	1	0	2	5	0	8	3	1	6	48

**Károly Róbert Főiskola**  
**Bemeneti teszt - kontroll csoport**

Ssz.	Név	Csop.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	Össz.
1.	Tanuló 1	0	6	6	10	0	9	0	9	5	8	1	0	10	64
2.	Tanuló 2	0	6	6	10	2	5	0	9	4	2	0	0	5	49
3.	Tanuló 3	0	6	4	10	0	2	2	4	0	0	3	1	7	39
4.	Tanuló 4	0	0	2	6	0	4	3	4	0	2	0	1	1	23
5.	Tanuló 5	0	6	6	7	0	0	0	7	0	0	0	1	0	27
6.	Tanuló 6	0	0	4	9	1	0	0	5	0	2	2	1	4	28
7.	Tanuló 7	0	6	3	10	0	4	0	1	0	0	2	0	10	36
8.	Tanuló 8	0	6	6	9	5	10	6	7	1	1	0	3	5	59
9.	Tanuló 9	0	6	3	5	1	7	5	9	4	8	0	5	0	53
10.	Tanuló 10	0	6	6	10	4	6	1	9	0	4	7	0	10	63
11.	Tanuló 11	0	6	6	10	3	7	0	7	0	8	1	1	10	59
12.	Tanuló 12	0	6	6	9	0	10	6	9	0	8	0	1	10	65
13.	Tanuló 13	0	6	4	10	0	6	0	6	0	4	2	1	5	44
14.	Tanuló 14	0	6	6	6	2	4	1	3	2	8	1	1	10	50
15.	Tanuló 15	0	6	0	8	0	2	1	3	0	2	0	1	10	33
16.	Tanuló 16	0	6	5	10	1	9	0	5	2	5	0	1	9	53
17.	Tanuló 17	0	6	2	7	1	8	2	5	2	0	0	3	2	38
18.	Tanuló 18	0	6	6	10	1	10	8	7	3	7	0	6	0	64
19.	Tanuló 19	0	6	3	10	0	6	0	4	0	5	2	3	4	43
20.	Tanuló 20	0	6	6	10	1	7	3	9	3	2	7	1	10	65
21.	Tanuló 21	0	6	5	10	1	4	2	8	0	3	1	2	6	48
22.	Tanuló 22	0	6	6	8	4	0	2	9	0	8	2	3	4	52
23.	Tanuló 23	0	3	6	9	1	10	2	6	7	8	1	0	0	53
24.	Tanuló 24	0	6	6	10	3	3	0	9	5	8	7	6	8	71
25.	Tanuló 25	0	6	6	10	1	7	0	6	4	8	0	1	10	59
26.	Tanuló 26	0	6	4	4	1	6	2	9	0	6	2	1	5	46
27.	Tanuló 27	0	6	6	9	0	10	4	9	0	4	0	0	0	48
28.	Tanuló 28	0	6	6	7	3	10	0	5	5	5	3	0	10	60
29.	Tanuló 29	0	6	2	10	3	10	5	4	3	8	3	1	10	65
30.	Tanuló 30	0	6	6	10	1	8	0	9	4	2	3	0	3	52
31.	Tanuló 31	0	6	5	10	3	2	1	6	2	8	6	0	8	57
32.	Tanuló 32	0	6	6	9	2	8	4	9	4	8	0	5	0	61
33.	Tanuló 33	0	6	6	6	1	6	0	6	0	4	2	1	10	48
34.	Tanuló 34	0	6	6	10	1	5	1	9	0	2	3	2	10	55
35.	Tanuló 35	0	6	6	8	2	7	0	5	2	2	7	0	10	55
36.	Tanuló 36	0	6	6	10	0	6	0	8	0	8	0	0	0	44
37.	Tanuló 37	0	6	6	10	2	8	4	3	6	7	5	1	9	67
38.	Tanuló 38	0	6	6	7	3	8	3	9	0	8	2	1	0	53

**Károly Róbert Főiskola**  
**Bemeneti teszt - kísérleti csoport**

Ssz.	Név	Csop.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	Össz.
1.	Tanuló 1	1	3	6	10	0	8	0	6	0	5	0	0	4	42
2.	Tanuló 2	1	0	4	8	2	6	1	4	0	8	0	0	0	33
3.	Tanuló 3	1	6	6	8	1	4	6	6	0	3	2	1	0	43
4.	Tanuló 4	1	6	6	7	2	2	1	1	0	2	1	1	1	30
5.	Tanuló 5	1	6	6	10	0	2	0	9	2	8	2	0	10	55
6.	Tanuló 6	1	0	4	9	2	6	0	4	0	8	0	1	10	44
7.	Tanuló 7	1	6	6	7	2	3	1	2	0	2	1	2	8	40
8.	Tanuló 8	1	6	6	10	0	10	5	9	5	8	0	1	0	60
9.	Tanuló 9	1	6	0	8	2	2	0	7	0	4	0	1	0	30
10.	Tanuló 10	1	6	6	8	0	10	5	9	0	8	7	0	5	64
11.	Tanuló 11	1	6	6	9	1	10	4	5	4	8	2	4	9	68
12.	Tanuló 12	1	6	6	10	3	2	3	7	0	2	4	1	7	51
13.	Tanuló 13	1	6	6	9	7	10	4	6	4	8	7	4	10	81
14.	Tanuló 14	1	6	6	9	0	7	3	6	0	8	1	0	0	46
15.	Tanuló 15	1	6	6	10	4	10	0	6	0	8	0	0	10	60
16.	Tanuló 16	1	6	6	9	1	8	0	6	3	0	3	0	9	51
17.	Tanuló 17	1	6	6	10	0	6	3	9	0	5	3	0	0	48
18.	Tanuló 18	1	6	6	8	0	7	3	5	0	0	0	0	0	35
19.	Tanuló 19	1	6	6	9	4	8	2	8	4	8	3	1	0	59
20.	Tanuló 20	1	6	6	10	1	10	4	9	4	8	2	0	0	60
21.	Tanuló 21	1	6	6	10	3	10	4	9	2	6	0	1	9	66
22.	Tanuló 22	1	6	6	10	1	6	1	9	1	1	1	1	10	53
23.	Tanuló 23	1	0	0	9	0	4	0	9	0	8	2	0	0	32
24.	Tanuló 24	1	6	6	9	1	10	0	8	0	8	4	1	0	53
25.	Tanuló 25	1	6	6	9	5	2	1	9	6	7	0	5	10	66
26.	Tanuló 26	1	6	6	10	0	6	0	0	0	5	1	1	2	37
27.	Tanuló 27	1	6	3	10	0	10	4	7	0	8	0	1	3	52

## F függelék – Kimeneti teszt

### Problémamegoldó feladatok

#### Kimeneti teszt

**Időtartam: 60 perc**

#### Adatok:

Név: ..... Osztály: .....

Életkor: ..... Legutóbbi, év végi Matematika jegye: .....

#### Feladatok:

1. Édesanya szilvás gombócot főzött ebédre. Anti nagyon éhes volt, megette hát a tál gombóc felét, Bea, a testvére meg a megmaradtak harmadrészét, s ezután 6 szilvás gombóc maradt a tálban. Hány szilvás gombóc készült? **6 pont**

*Válasz:*

*Indoklás:*

2. Egy apa 1600 koronát hagyott három fiára. A végrendeletben meghagyta, hogy a legidősebb fiú jussa 200 koronával több legyen a középsőénél, a középsőé pedig 100 koronával több, mint a legkisebbé. Számítsuk ki mindegyikük részét. **6 pont**

*Válasz:*

*Indoklás:*

3. Nekeresd országban él két különböző ember: Péter, aki minden hétfőn, kedden és szerdán hazudik, máskor igazat mond, és Pál, aki minden csütörtökön, pénteken, szombaton hazudik, és a többi napon igazat mond. Egyik napon ezt állították:

**8 pont**

Péter: Tegnap hazudós napom volt.

Pál: Tegnap nekem is hazudós napom volt.

A hét melyik napján volt ez?

*Válasz:*

*Indoklás:*

4. Gyermektábort szervez az iskola. 46 gyerek (26 lány és 20 fiú) jelentkezett a táborba, és 8 felnőtt (4 férfi és 4 nő) vállalta, hogy elkíséri a gyerekeket, és vezeti a tábort.

**10 pont**

Felnőttek	Szoba	Férőhely
Kiss Éva	1.101	12
Nagy Lászlóné	1.102	8
Szabóné Nagy Ágota	1.103	8
Kovács Józsefné	1.104	8
Gábor Zoltán	2.201	8
Keckskeméti Róbert	2.202	6
Tóth István	2.203	6
Kelemen Tamás		

**Hálótermi szabályok:**

1. A fiúk és a lányok külön hálóteremben alszanak.
2. Minden hálóteremben legalább egy felnőttnek kell aludnia.
3. Az egy szobában alvó felnőtt(ek)nek azonos nemű(ek)nek kell lenni(ük) a gyerekekkel.

A táblázat kitöltésével helyezze el a hálótermekben a 46 gyereket és a 8 felnőttet úgy, hogy valamennyi szabályt figyelembe veszi!

Szoba	Lányok száma	Fiúk száma	Felnőtt neve
1.101			
1.102			
1.103			
1.104			
2.201			
2.202			
2.203			

5. Manócska egyszer madártejet készített barátainak. Mielőtt azonban tálalhatta volna, a madártej eltűnt. A tőkházban Manócskán kívül még négyen voltak : Mazsola, Tádé, Cica-Mica és Morzsi. A kérdésre, hogy ki ette meg a madártejet, így válaszoltak:

**8 pont**

Mazsola: Tádé volt. ....  
Tádé: Cica-Mica volt. ....  
Morzsi: Nem én voltam. ....  
Cica-Mica: Tádé fillent. ....

Ki ette meg a madártejet, ha tudjuk, hogy négyőjük közül pontosan egy hazudott és csak egy ehette meg a madártejet. Írd a nevek mellé, hogy hazudott-e, vagy igazat mondott!

*Válasz:*

*Indoklás:*

6. A 7 törpe házikójában valaki eltört egy tányért. Hófehérkének így számoltak be a történetről:

**7 pont**

1. állítás

2. állítás

Tudor: Nem Szundi volt. Én voltam.

.....

.....

Morgó: Nem én voltam. Nem Hapci volt.

.....

.....

Vidor: Tudor volt. Nem Morgó volt.

.....

.....

Ki törte el a tányért, ha a törpék egyik állítása igaz, a másik hamis?

Válasz: .....

Írd a mondatok mellé, hogy melyik állítása igaz és melyik hamis az egyes törpéknek!

Indoklás:

7. Energiaszükséglet

**10 pont**

E feladat lényege, hogy olyan ételeket kell kiválasztani, amelyek Zedország egy lakójának energiaszükségletéhez igazodnak. A következő táblázat azt mutatja, hogy különböző emberek esetében mennyi az ajánlott energiaszükséglet kilojoule-ban kifejezve (kJ).

AZ AJÁNLOTT NAPI ENERGIA SZÜKSÉGLET FELNÖTTEK SZÁMÁRA			
Életkor (években)	Aktivitási szint	FÉRFIAK	NŐK
		Szükséges energiamennyiség (kJ)	Szükséges energiamennyiség (kJ)
18-tól 29-ig	Könnyű	10660	8360
	Mérsékelt	11080	8780
	Nehéz	14420	9820
30-tól 59-ig	Könnyű	10450	8570
	Mérsékelt	12120	8990
	Nehéz	14210	9790
60 és afölött	Könnyű	8780	7500
	Mérsékelt	10240	7940
	Nehéz	11910	8780

EGYES FOGLALKOZÁSOKNAK MEGFELELŐ AKTIVITÁSI SZINT		
<b>Könnyű:</b>	<b>Mérsékelt:</b>	<b>Nehéz:</b>
Bolti eladó	Tanár	Építőmunkás
Irodai dolgozó	Üzletkötő	Fizikai munkás
Háztartásbeli	Ápoló	Sportoló

Molnár Károly 45 éves tanár. Mennyi az ajánlott napi energiaszükséglete kJ-ban?

Válasz: ..... kilojoule.

Szalay Edit 19 éves magasugró. Egy este néhány barátja étterembe hívja Editet vacsorázni. Íme az étlap:

<b>ÉTLAP</b>		<b>1 adag energiatartalma Edit becslése szerint (kJ)</b>
<b>Levesek:</b>	Paradicsomleves	355
	Gombakrémleves	585
<b>Főételek:</b>	Mexikói csirke	960
	Karibi gyömbéres csirke	795
	Zsályás sertéspörkölt	920
<b>Saláták:</b>	Burgonyasaláta	750
	Vegyes zöldségsaláta	335
	Kuszkusz saláta	480
<b>Desszertek:</b>	Almás-málnás lepény	1380
	Túrótorta	1005
	Répatorta	565
<b>Tejes turmixok:</b>	Csokoládé	1590
	Vanília	1470

Az étteremben egységáras menü is kapható.

<p style="text-align: center;"><b>Menü</b>  <b>ára 50 zed</b>  Paradicsomleves  Karibi gyömbéres csirke  Répatorta</p>
--

Edit minden nap feljegyzi, hogy mit evett. Az aznap vacsora előttig elfogyasztott ételek 7520 kJ energiabevitelnek felelnek meg.

Edit azt szeretné elérni, hogy az összes energiabevitale 500 kJ-nál többel **ne haladja meg vagy ne maradjon alatta a számára ajánlott napi mennyiségnek.**

Dönts el, hogy vajon az egységáras menü elfogyasztásával Edit a napi ajánlott energiaszükségletéhez képest a  $\pm 500$  kJ-os határon belül marad-e! Úgy dolgozz, hogy a munkád nyomonkövethető legyen!

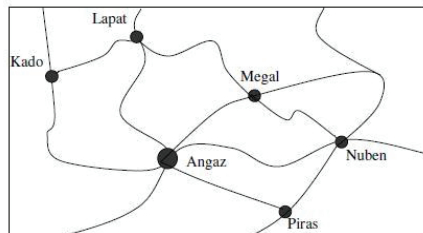
8. Vakáció

9 pont

Ebben a feladatban egy nyaralás útvonalát kell megtervezned.

Az 1. és 2. ábrán az adott terület térképe és a városok közötti távolságok láthatók.

1. ábra: Térkép a városokat összekötő útvonalakról



2. ábra: Az egyes városok közötti legrövidebb útvonal kilométerben

Angaz						
Kado	550					
Lapat	500	300				
Megal	300	850	550			
Nuben	500		1000	450		
Piras	300	850	800	600	250	
	Angaz	Kado	Lapat	Megal	Nuben	Piras

Számítsd ki, hogy milyen hosszú a legrövidebb útvonal Nuben és Kado között!

Útvonal hossza: ..... kilométer.

Zita Angazban lakik. Szeretne ellátogatni Kadoba és Lapatba. Egy nap alatt legfeljebb 300 kilométert tud megtenni, de megszakíthatja utazását, és éjszakára bárhol letáborozhat két város között.

Zita két éjszakát akar tölteni mindkét városban úgy, hogy egy teljes napja legyen városnézésre mindkét helyen.

Az alábbi táblázat kitöltésével mutasd be Zita útvonalát! Írd be, hogy hol tölti az éjszakákat!

Nap	Éjszakai szálláshely
1	Kemping Angaz és Kado között.
2	
3	
4	
5	
6	
7	Angaz



9. Három ládikó mindegyikében 2-2 golyó van: fehér-fehér, piros-piros, fehér-piros, és mindegyik ládán felirat: FF, PP, FP, ami a ládában lévő golyók színét jelzi.

Azonban mindegyik felirat hamis.

**10 pont**

Valamelyik - általad választott - ládikóból kivehetsz egy golyót, s ezek után meg kell mondanod, hogy az egyes ládikókban milyen golyók vannak. Hogyan csinálod ezt?

FF

PP

FP

10. Egy műszaki főiskola az alábbi 12 tantárgyat kínálja egy olyan 3 féléves képzés keretében, amely során minden tantárgyat egy félévig tanítanak. **10 pont**

	Tantárgy kódja	Tantárgy neve
1	M1	Mechanika, 1-es szint
2	M2	Mechanika, 2-es szint
3	E1	Elektronika, 1-es szint
4	E2	Elektronika, 2-es szint
5	Ü1	Üzleti ismeretek, 1-es szint
6	Ü2	Üzleti ismeretek, 2-es szint
7	Ü3	Üzleti ismeretek, 3-as szint
8	Sz1	Számítógépes ismeretek, 1-es szint
9	Sz2	Számítógépes ismeretek, 2-es szint
10	Sz3	Számítógépes ismeretek, 3-as szint
11	T1	Technológia és adatkezelés, 1-es szint
12	T2	Technológia és adatkezelés, 2-es szint

Minden diák 4 tantárgyat vesz fel egy félévben, így a 12 tantárgyat 3 félév alatt sajátítja el. Egy diák csak akkor veheti fel egy tantárgy magasabb szintű kurzusát, ha az alsóbb szintet ebből a tárgyból már előzőleg elvégezte. Az Üzleti ismeretek 3-at például csak akkor veheti fel, ha befejezte az Üzleti ismeretek 1-et és 2-t. Ezen kívül az Elektronika 1-et csak a Mechanika 1 után lehet felvenni, és az Elektronika 2-t csak a Mechanika 2 után.

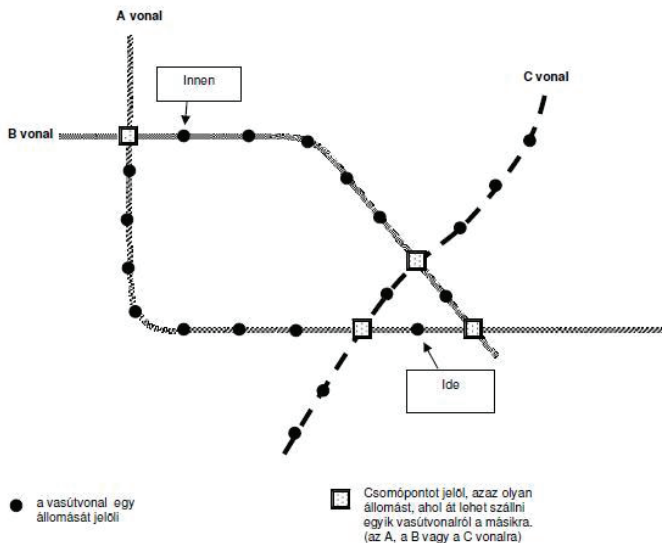
Döntse el, hogy melyik félévben mely tantárgyakat hirdessék meg, és töltsé ki a táblázatot! A tantárgyak kódját írja a táblázatba!

	1. tantárgy	2. tantárgy	3. tantárgy	4. tantárgy
1. félév				
2. félév				
3. félév				

# 11. Közlekedés

8 pont

A következő ábra egy zedországbeli város közlekedési hálózatának egy részletét mutatja. A három vasútvonalon kívül azt is megjelöltük, hogy hol állsz most, és hová kell eljutnod.



Az útiköltség az érintett állomások számától függ (nem számítva azt az állomást, ahonnan indulsz). Minden érintett állomás 1 zeddel drágítja a jegyet.

A menetidő két szomszédos állomás között körülbelül 2 perc.

A csomópontokon az átszállás egyik vonalról a másikra körülbelül 5 percet vesz igénybe.

Az ábrán látható az az állomás, ahol jelenleg állsz („Innen”), illetve az az állomás ahova el kell jutnod („Ide”). **Jelöld meg az ábrán** az utazásra fordított idő és az útiköltség szempontjából legkedvezőbb útvonalat, és állapítsd meg az útiköltséget és az utazás hozzávetőleges idejét!

Útiköltség: .....zed.

Az utazás hozzávetőleges ideje: .....perc.

12. Négy lány futóversenyen vett részt, melyen nem volt holtverseny.  
A verseny után mindegyiket megkérdezték, melyik helyen végzett.

**8 pont**

Anna: „Nem lettem sem első, sem utolsó.” .....  
Bella: „Nem lettem első.” .....  
Csilla: „Első lettem.” .....  
Dóra: „Én lettem az utolsó.” .....

Valaki, aki a versenyt is látta, ezt mondta: „A négy válasz közül három igaz, egy hamis.”

Ki mondott valótlant?.....

Írd a nevek mellé, hogy hazudott-e, vagy igazat mondott!

Ki volt az első?.....

*Indoklás:*

## G függelék – A kimeneti tesztek eredményei

### József Attila Szakközépiskola, Szakiskola és Kollégium

#### Kimeneti teszt - kontroll csoport

Ssz.	Név	Csop.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	Össz.
1.	Tanuló 1	0	0	0	0	4	2	0	1	3	0	0	0	0	10
2.	Tanuló 2	0	0	0	0	0	2	0	2	3	0	0	0	0	7
3.	Tanuló 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4.	Tanuló 4	0	4	2	0	0	2	0	0	0	0	0	0	2	10
5.	Tanuló 5	0	0	0	2	9	0	0	3	3	0	0	0	0	17
6.	Tanuló 6	0	0	2	1	6	0	0	2	1	0	0	2	2	16
7.	Tanuló 7	0	6	0	3	0	0	0	1	0	0	0	7	0	17
8.	Tanuló 8	0	0	0	0	1	0	0	0	0	0	0	0	4	5
9.	Tanuló 9	0	0	0	4	4	3	0	2	0	0	0	0	4	17
10.	Tanuló 10	0	0	0	0	2	0	0	1	3	0	1	0	1	8
11.	Tanuló 11	0	0	0	0	0	0	0	2	5	0	5	0	0	12
12.	Tanuló 12	0	0	0	0	2	0	0	0	0	0	0	0	0	2
13.	Tanuló 13	0	6	4	5	4	2	0	2	0	0	0	5	2	30
14.	Tanuló 14	0	0	3	2	1	0	0	0	0	0	4	0	2	12
15.	Tanuló 15	0	1	3	1	10	0	0	2	0	0	6	0	0	23
16.	Tanuló 16	0	0	0	0	0	0	0	3	0	0	0	0	0	3
17.	Tanuló 17	0	0	0	0	0	0	0	2	0	0	0	3	0	5
18.	Tanuló 18	0	0	0	0	0	0	0	2	0	0	0	3	2	7
19.	Tanuló 19	0	0	2	0	1	2	0	3	0	0	8	0	2	18
20.	Tanuló 20	0	0	3	0	3	0	0	2	0	0	0	0	2	10
21.	Tanuló 21	0	6	0	4	6	0	0	10	3	0	0	0	2	31
22.	Tanuló 22	0	0	2	0	0	0	0	3	0	0	0	0	0	5
23.	Tanuló 23	0	0	0	0	1	0	0	3	0	0	0	0	3	7
24.	Tanuló 24	0	0	0	2	0	0	0	1	0	0	0	3	4	10
25.	Tanuló 25	0	0	0	2	0	0	0	2	0	0	0	2	0	6
26.	Tanuló 26	0	4	2	1	4	0	0	3	3	0	8	5	2	32
27.	Tanuló 27	0	0	0	4	0	0	0	1	0	0	0	5	2	12
28.	Tanuló 28	0	4	3	1	10	0	0	3	3	0	0	0	0	24

**József Attila Szakközépiskola, Szakiskola és Kollégium**  
**Kimeneti teszt - kísérleti csoport**

Ssz.	Név	Csop.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	Össz.
1.	Tanuló 1	1	4	0	0	10	3	0	3	2	0	3	0	2	27
2.	Tanuló 2	1	3	0	0	0	2	0	10	3	0	0	0	0	18
3.	Tanuló 3	1	0	2	0	10	0	0	4	3	2	1	0	2	24
4.	Tanuló 4	1	0	0	0	0	0	0	0	0	0	0	0	2	2
5.	Tanuló 5	1	0	0	0	1	0	0	3	0	0	2	3	0	9
6.	Tanuló 6	1	0	2	0	10	0	2	5	3	2	4	0	0	28
7.	Tanuló 7	1	6	0	0	0	2	0	2	0	0	1	3	0	14
8.	Tanuló 8	1	4	2	2	9	0	0	7	3	0	0	3	3	33
9.	Tanuló 9	1	6	4	4	10	4	0	3	0	2	6	4	2	45
10.	Tanuló 10	1	0	0	0	0	0	0	2	2	0	0	0	2	6
11.	Tanuló 11	1	0	0	0	1	0	0	0	1	0	3	0	2	7
12.	Tanuló 12	1	6	3	4	8	4	0	10	9	0	8	0	6	58
13.	Tanuló 13	1	5	3	4	10	0	0	2	6	0	8	3	4	45
14.	Tanuló 14	1	0	3	2	1	0	0	4	2	0	4	0	4	20
15.	Tanuló 15	1	0	1	0	0	2	0	0	0	0	0	0	0	3
16.	Tanuló 16	1	4	3	3	0	1	0	2	4	0	8	0	3	28
17.	Tanuló 17	1	6	4	0	10	0	0	3	3	0	10	3	2	41
18.	Tanuló 18	1	2	2	0	0	0	0	3	9	0	10	5	2	33

## Vak Bottyán János Katolikus Műszaki és Közgazdasági Középiskola

### Kimeneti teszt - kontroll csoport

Ssz.	Név	Csop.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	Össz.
1.	Tanuló 1	0	6	0	0	0	2	0	10	9	7	0	0	4	38
2.	Tanuló 2	0	0	0	0	3	0	0	7	2	0	9	3	2	26
3.	Tanuló 3	0	6	6	4	10	1	2	8	9	0	10	0	4	60
4.	Tanuló 4	0	0	0	3	0	0	0	2	0	0	0	0	0	5
5.	Tanuló 5	0	6	6	0	10	2	0	8	0	0	10	0	0	42
6.	Tanuló 6	0	5	6	4	10	1	2	10	8	0	10	8	4	68
7.	Tanuló 7	0	0	2	1	10	1	0	10	0	0	10	0	4	38
8.	Tanuló 8	0	5	6	4	10	0	2	10	8	0	0	3	4	52
9.	Tanuló 9	0	4	6	4	6	2	4	10	9	0	8	5	4	62
10.	Tanuló 10	0	0	6	2	3	2	2	2	3	0	0	0	4	24
11.	Tanuló 11	0	4	2	3	10	0	0	8	9	0	0	0	0	36
12.	Tanuló 12	0	4	2	3	4	0	0	0	3	0	0	0	0	16
13.	Tanuló 13	0	5	2	0	1	2	0	5	3	0	0	0	2	20
14.	Tanuló 14	0	5	6	2	0	2	0	8	0	0	10	0	4	37
15.	Tanuló 15	0	4	1	1	10	2	0	8	2	0	4	3	2	37
16.	Tanuló 16	0	3	2	3	10	2	0	2	3	0	0	0	4	29
17.	Tanuló 17	0	6	2	3	10	2	2	2	7	0	0	3	2	39
18.	Tanuló 18	0	0	2	0	0	3	2	2	3	0	0	8	3	23
19.	Tanuló 19	0	0	6	1	10	2	2	10	3	0	0	0	0	34
20.	Tanuló 20	0	3	1	4	6	2	0	10	9	0	0	3	4	42
21.	Tanuló 21	0	0	1	0	1	0	0	2	5	0	9	3	2	23
22.	Tanuló 22	0	2	2	2	10	0	0	1	6	0	2	0	2	27
23.	Tanuló 23	0	5	1	3	0	1	2	2	3	10	0	3	1	31
24.	Tanuló 24	0	4	6	4	7	1	0	10	9	3	0	0	0	44
25.	Tanuló 25	0	6	0	0	10	0	3	2	3	0	8	0	0	32
26.	Tanuló 26	0	4	6	3	2	0	2	7	7	0	0	3	0	34
27.	Tanuló 27	0	0	1	3	10	2	0	3	9	0	5	2	4	39
28.	Tanuló 28	0	0	0	0	0	2	0	2	0	0	3	0	2	9
29.	Tanuló 29	0	4	3	3	8	2	0	8	3	0	0	0	2	33
30.	Tanuló 30	0	0	0	3	0	2	0	9	0	0	0	3	0	17
31.	Tanuló 31	0	3	6	3	10	0	1	8	3	0	10	0	4	48
32.	Tanuló 32	0	6	2	3	6	2	0	8	5	0	10	0	4	46

**Vak Bottyán János Katolikus Műszaki és Közgazdasági Középiskola**  
**Kimeneti teszt - kísérleti csoport**

Ssz.	Név	Csop.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	Össz.
1.	Tanuló 1	1	6	6	3	10	4	1	10	9	7	10	8	8	82
2.	Tanuló 2	1	4	4	0	9	0	0	10	4	0	0	0	2	33
3.	Tanuló 3	1	5	6	3	10	4	1	9	9	7	10	8	8	80
4.	Tanuló 4	1	6	6	3	10	5	4	10	9	10	10	3	5	81
5.	Tanuló 5	1	6	6	1	10	2	0	10	9	0	10	4	4	62
6.	Tanuló 6	1	4	6	3	10	0	0	10	9	4	9	3	0	58
7.	Tanuló 7	1	1	0	2	4	0	0	6	0	0	5	3	2	23
8.	Tanuló 8	1	5	6	3	10	0	0	8	9	0	9	2	5	57
9.	Tanuló 9	1	4	2	3	2	0	0	8	5	0	0	6	0	30
10.	Tanuló 10	1	6	6	4	3	8	1	10	7	0	9	1	0	55
11.	Tanuló 11	1	5	4	2	4	3	3	10	0	5	10	8	0	54
12.	Tanuló 12	1	1	6	1	10	2	2	8	9	0	10	6	4	59
13.	Tanuló 13	1	6	6	3	9	3	1	10	3	1	4	0	5	51
14.	Tanuló 14	1	6	6	2	10	2	0	9	6	0	3	3	2	49
15.	Tanuló 15	1	5	5	4	10	2	0	3	5	0	0	3	0	37
16.	Tanuló 16	1	2	6	1	6	2	0	10	4	0	0	0	0	31
17.	Tanuló 17	1	4	6	1	10	4	1	10	7	0	0	0	4	47
18.	Tanuló 18	1	6	0	1	6	2	0	7	4	0	10	3	4	43
19.	Tanuló 19	1	1	0	1	1	0	0	6	6	0	0	0	2	17
20.	Tanuló 20	1	4	2	3	1	0	0	2	0	0	0	2	0	14
21.	Tanuló 21	1	4	6	0	2	0	0	10	9	0	8	0	0	39
22.	Tanuló 22	1	6	2	3	0	0	0	10	0	1	5	5	0	32
23.	Tanuló 23	1	2	0	1	0	2	0	0	0	0	1	0	2	8
24.	Tanuló 24	1	1	3	1	0	2	0	2	3	0	10	3	2	27
25.	Tanuló 25	1	5	1	1	2	2	0	3	3	0	8	3	0	28
26.	Tanuló 26	1	4	6	4	10	5	2	10	8	0	0	0	3	52

**Károly Róbert Főiskola**  
**Kimeneti teszt - kontroll csoport**

Ssz.	Név	Csop.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	Össz.
1.	Tanuló 1	0	6	6	6	10	8	0	10	9	10	10	5	8	88
2.	Tanuló 2	0	6	6	3	10	8	2	10	9	3	6	0	4	67
3.	Tanuló 3	0	4	6	0	10	0	0	0	2	0	7	5	4	38
4.	Tanuló 4	0	1	2	1	3	2	0	3	6	5	7	3	1	34
5.	Tanuló 5	0	2	0	0	0	0	0	2	0	0	0	0	0	4
6.	Tanuló 6	0	4	6	4	10	2	0	10	9	1	10	3	6	65
7.	Tanuló 7	0	6	6	0	10	1	3	6	9	1	0	8	6	56
8.	Tanuló 8	0	4	0	4	10	0	0	4	9	1	10	3	4	49
9.	Tanuló 9	0	6	6	4	0	2	0	2	6	0	10	0	2	38
10.	Tanuló 10	0	6	3	3	10	2	2	10	4	0	10	3	0	53
11.	Tanuló 11	0	6	1	0	10	0	2	10	9	0	9	0	2	49
12.	Tanuló 12	0	0	2	1	5	2	0	2	2	0	9	3	3	29
13.	Tanuló 13	0	3	0	4	7	2	0	10	9	0	9	7	0	51
14.	Tanuló 14	0	4	0	3	3	2	0	3	0	0	1	3	0	19
15.	Tanuló 15	0	6	6	3	10	8	7	10	9	8	10	3	8	88
16.	Tanuló 16	0	3	0	1	2	2	0	4	2	0	5	3	3	25
17.	Tanuló 17	0	0	0	0	3	0	0	2	5	0	0	0	0	10
18.	Tanuló 18	0	2	6	1	1	1	0	10	3	0	10	5	0	39
19.	Tanuló 19	0	2	6	4	10	6	5	10	0	10	0	3	8	64
20.	Tanuló 20	0	6	3	4	3	3	0	8	9	5	10	0	6	57
21.	Tanuló 21	0	4	6	3	10	0	0	10	9	1	10	0	4	57
22.	Tanuló 22	0	0	2	0	9	2	0	2	3	0	10	3	4	35
23.	Tanuló 23	0	4	4	3	10	8	0	10	9	3	10	0	6	67
24.	Tanuló 24	0	0	2	1	2	2	0	10	9	0	6	3	2	37
25.	Tanuló 25	0	6	6	0	2	0	0	2	9	3	10	3	4	45
26.	Tanuló 26	0	4	6	3	10	2	0	10	3	0	9	3	4	54
27.	Tanuló 27	0	3	1	1	10	2	2	10	9	0	10	5	4	57
28.	Tanuló 28	0	5	6	3	9	2	2	10	9	2	10	0	0	58
29.	Tanuló 29	0	5	6	3	8	0	0	7	9	0	9	8	4	59
30.	Tanuló 30	0	5	2	3	10	3	2	10	9	1	9	4	5	63
31.	Tanuló 31	0	0	6	1	4	1	2	8	9	0	0	4	6	41
32.	Tanuló 32	0	0	0	3	4	2	0	9	9	0	1	6	2	36



**Károly Róbert Főiskola**  
**Kimeneti teszt - kísérleti csoport**

Ssz.	Név	Csop.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	Össz.
1.	Tanuló 1	1	6	6	3	10	4	1	10	0	0	10	2	6	58
2.	Tanuló 2	1	4	6	4	0	2	0	10	9	0	10	8	2	55
3.	Tanuló 3	1	0	3	1	10	0	1	10	3	0	10	0	0	38
4.	Tanuló 4	1	4	3	1	10	1	0	2	0	0	10	5	0	36
5.	Tanuló 5	1	6	0	0	2	0	0	10	6	0	10	3	0	37
6.	Tanuló 6	1	4	0	3	2	1	2	3	0	0	5	3	4	27
7.	Tanuló 7	1	4	6	4	8	4	1	10	9	1	10	8	3	68
8.	Tanuló 8	1	5	3	4	9	4	0	10	9	7	10	8	0	69
9.	Tanuló 9	1	2	0	1	2	2	0	2	9	0	0	0	0	18
10.	Tanuló 10	1	4	6	4	0	2	0	7	3	1	0	2	1	30
11.	Tanuló 11	1	6	6	6	5	3	1	10	4	0	10	0	5	56
12.	Tanuló 12	1	4	1	0	10	2	0	10	9	0	0	3	5	44
13.	Tanuló 13	1	6	5	5	10	6	0	10	9	0	10	3	6	70
14.	Tanuló 14	1	4	6	3	10	2	2	2	0	0	10	3	0	42
15.	Tanuló 15	1	6	6	4	10	8	0	10	3	0	1	5	0	53
16.	Tanuló 16	1	0	6	1	2	2	0	8	4	0	10	3	5	41
17.	Tanuló 17	1	6	4	4	10	0	0	8	7	0	10	3	0	52
18.	Tanuló 18	1	5	3	1	3	2	0	10	3	1	10	2	4	44
19.	Tanuló 19	1	4	1	3	10	4	3	3	0	10	10	3	5	56
20.	Tanuló 20	1	4	4	2	10	4	3	10	3	0	10	2	4	56
21.	Tanuló 21	1	6	6	4	10	5	7	10	9	10	10	2	6	85
22.	Tanuló 22	1	4	6	4	4	3	0	10	6	0	9	6	5	57
23.	Tanuló 23	1	6	5	1	10	1	0	5	3	7	10	0	6	54
24.	Tanuló 24	1	2	5	4	1	0	1	10	3	0	10	2	0	38
25.	Tanuló 25	1	6	6	4	0	3	0	10	0	10	10	4	3	56
26.	Tanuló 26	1	0	4	1	5	1	0	10	0	0	10	3	2	36
27.	Tanuló 27	1	4	6	6	0	8	3	8	3	1	2	5	7	53
28.	Tanuló 28	1	5	6	4	9	5	5	10	9	0	10	3	4	70
29.	Tanuló 29	1	6	5	4	10	4	3	10	6	0	10	8	6	72

## H függelék – Középiskoláknak kiküldött kérdőív

### KÉRDŐÍV

Kérem a választási lehetőségek közül emelje ki, amelyek az Ön iskolájára illik.

1. Iskola típusa:

Gimnázium, Szakközépiskola, Egyéb: .....

2. Iskola helyszíne:

Főváros, Megyeszékhely, Város, Egyéb: .....

3. Tanulnak-e intézményükben tanulók speciális, informatikai specializációval rendelkező osztályban? Igen – Nem

4. Ha igen, mi a specializáció neve:

.....

5. Folytatnak-e valamilyen informatikai OKJ-s képzést, vagy Felsőfokú szakképzést intézményükben? Igen – Nem

6. Ha igen, mi a képzés neve, azonosítója:

.....

.....

7. Tanítják-e a Prolog logikai programnyelvet? Igen –Nem

8. Ha igen, akkor milyen képzésben, évfolyamon, óraszámban?

.....

.....

9. Milyen egyéb programnyelveket tanítanak az iskolában? A táblázat, ha szükséges, bővíthető.

Ssz.	Programnyelv	Képzés	Évfolyam	Összes óraszám
1.				
2.				
3.				
4.				
5.				

10. Milyen képzésben, évfolyamon, óraszámban tanítanak adatbáziskezelést?

.....  
.....

11. Tanítanak-e SQL-t? Igen – Nem

12. Ha igen, akkor milyen képzésben, évfolyamon, óraszámban?

.....

13. Működtetnek-e az iskolában e-learning rendszert: Igen – Nem

14. Ha igen, melyet?

pl.: Moodle, Ilias, Drupal, Egyéb: .....

15. Kb. hány működő kurzus van az e-learning rendszerben?

.....

16. Kb. hány felhasználót szolgál ki a rendszer?

.....

17. Tanítanak-e valamilyen formában mesterséges intelligenciát? Igen – Nem

18. Ha igen, akkor milyen képzésben, évfolyamon, óraszámban,  
milyen tartalommal?

.....

19. Milyen szervezett módokon fejlesztik a tanulók problémamegoldó  
képességét?

.....

.....

Köszönöm választát!

Kérem juttassa el ezt a dokumentumot elektronikusan az  
[rpantva@karolyrobert.hu](mailto:rpantva@karolyrobert.hu) címre!

Tisztelettel és köszönettel: Pántya Róbert